Brain-Computer Interface with cortical electrical activity recording

Andriy Yelisyeyev

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Modèles, méthodes et algorithmes pour la Biologie, la Santé et l'Environnement**

Arrêté ministériel : 7 août 2006

Présentée par

Andriy YELISYEYEV

Thèse dirigée par **Tetiana AKSENOVA**

préparée au sein du **CEA/LETI/CLINATEC**
dans **l'École Doctorale « Ingénierie pour la Santé , la Cognition et l'Environnement »**

Interface cerveau-machine à partir d'enregistrement électrique cortical

Thèse soutenue publiquement le **8.12.2011**,
devant le jury composé de :

Prof. François BERGER
Président
Prof. Roman BORISYUK
Rapporteur
Prof. Martin BOGDAN
Rapporteur
Dr. Tetiana AKSENOVA
Membre
Prof. Alim-Louis BENABID
Membre
Prof. Laura SACERDOTE
Membre

UNIVERSITÉ DE GRENOBLE

Ph. D. Thesis

To obtain the degree of

DOCTOR OF THE UNIVERSITY OF GRENOBLE

Specialties: **Models, methods and algorithms for Biology, Health and Environment**

Ministerial Order: 7 August 2006

Presented by

Andriy YELISYEYEV

Thesis directed by **Tetiana AKSENOVA**

Prepared within **CEA/LETI/CLINATEC**
in **l'École Doctorale Ingénierie pour la santé la Cognition et l'Environnement**

Brain-Computer Interface with cortical electrical activity recording

Thesis defended **8.12.2011**
Jury:
Prof. François BERGER
President
Prof. Roman BORISYUK
Rapporteur
Prof. Martin BOGDAN
Rapporteur
Dr. Tetiana AKSENOVA
Member
Prof. Alim-Louis BENABID
Member
Prof. Laura SACERDOTE
Member

Résumé

Une Interface Cerveau-Machine (ICM) est un système permettant de transformer l'activité neurale du cerveau en une commande d'effecteurs externes. Cette étude correspond à une étape vers une ICM totalement autonome fonctionnant dans un environnement naturel ce qui est d'une importance cruciale pour les futures applications cliniques d'une ICM. Pour représenter l'environnement naturel, des expériences avec une ICM binaire asynchrone ont été réalisées avec des animaux libres de se mouvoir. En comparaison avec les études précédentes, des expériences sur le long terme ont été réalisées, ce qui est plus conforme aux exigences des applications de la vie réelle. L'objectif principal de cette étude est de différencier le modèle spécifique neuronal lié à l'intention d'action de l'activité de fond du cerveau chez des animaux libres de tous mouvements. Pour atteindre le niveau nécessaire de sélectivité, l'analyse Multi-Voies PLS a été choisie sachant qu'elle fournit simultanément un traitement du signal dans plusieurs domaines, à savoir, temporel, fréquentiel et spatial. Pour améliorer la capacité de l'approche générique Multi-Voies PLS pour le traitement de données à grandes dimensions, l'algorithme « Itérative NPLS » est introduit dans notre travail. En ayant des besoins plus faibles en mémoire, cet algorithme fournit des traitements de grands ensembles de données, permet une résolution élevée, préserve l'exactitude de l'algorithme générique et démontre une meilleure robustesse. Pour la calibration adaptative d'un système ICM, l'algorithme récursif NPLS est proposé. Finalement, l'algorithme pénalisé NPLS est développé pour la sélection efficace d'un sous-ensemble de fonctions, à savoir, un sous-ensemble d'électrodes. Les algorithmes proposés ont été testés sur des ensembles de données artificielles et réelles. Ils ont démontré une performance qui est comparable à celle d'un algorithme générique NPLS. Leur efficacité de calcul est acceptable pour les applications en temps réel. Les algorithmes développés ont été appliqués à la calibration d'un système ICM et ont été utilisés dans des expériences d'ICM avec bouclage en temps réel chez des animaux. Enfin, les

méthodes proposées représentent une approche prospective pour de futurs développements de systèmes ICM humains.

Mots-clés : Interface Cerveau-Machine, Electrocorticographie, Traitement du signal, Multi-voies analyse, Modélisation adaptative, Ondelettes.

Abstract

Brain Computer Interface (BCI) is a system for translation of brain neural activity into commands for external devices. This study was undertaken as a step toward the fully autonomous (self-paced) BCI functioning in natural environment which is of crucial importance for BCI clinical applications. To model the natural environment binary self-paced BCI experiments were carried out in freely moving animals. In comparison to the previous works, the long-term experimental sessions were carried out, which better comply with the real-life applications requirements. The main goal of the study was to discriminate the specific neuronal pattern related to the animal's control action against background brain activity of freely-moving animal. To achieve the necessary level of selectivity the Multi-Way Analysis was chosen since it provides a simultaneous signal processing in several domains, namely, temporal, frequency and spatial. To improve the capacity of the generic Multy-Way Partial Least Squares (NPLS) approach for treatment of high-dimensional data, the Iterative NPLS algorithm is introduced in the current study. Having lower memory requirements it provides huge datasets treatment, allows high resolution, preserves the accuracy of the generic algorithm, and demonstrates better robustness. For adaptive calibration of BCI system the Recursive NPLS algorithm is proposed. Finally, the Penalized NPLS algorithm is developed for effective selection of feature subsets, namely, for subset of electrodes. The proposed algorithms were tested on artificial and real datasets. They demonstrated performance which either suppress or is comparable with one of the generic NPLS algorithm. Their computational efficiency is acceptable for the real-time applications. Developed algorithms were applied for calibration of the BCI system and were used in the real-time close-loop binary BCI experiments in animals. The proposed methods represent a prospective approach for further development of a human BCI system.

Keywords: Brain–Computer Interface, Electrocorticography, Signal Processing, Multi-Way Analysis, Adaptive Modeling, Wavelets.

TABLE OF CONTENTS

Chapter 1. FUNCTIONAL BRAIN COMPUTER INTERFACE

1.1 Introduction

Brain Computer Interface (BCI), also known as *Brain Interface* (BI), and *Brain Machine Interface* (BMI), is a system for translating the brain neural activity into commands to external devices (Wolpaw et al., 2002). In other words, BCI aims to provide an alternative communication pathway for subjects to interact with their environment without using muscles. Such a system could be helpful for people suffering from severe motor disabilities to control wheelchairs, prostheses, etc. High-level general scheme of BCI system is shown in Figure 1.1.

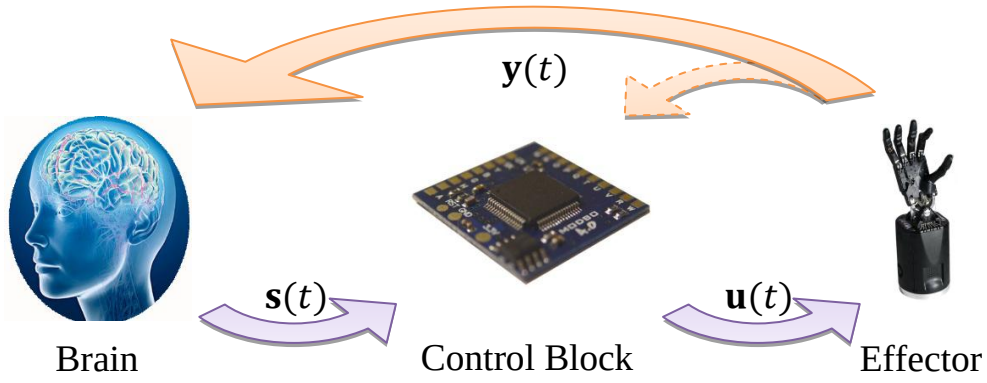

Figure 1.1 General scheme of BCI system. Here $\mathbf{s}(t)$ is a vector reflecting brain neural activity, $\mathbf{u}(t)$ is a vector of control block command, and $\mathbf{y}(t)$ represents the state of the effector at the moment t. The dotted arrow in the figure represents the connection which is active only at a certain period of the BCI's operation cycle.

Indeed, the operation cycle of a BCI system contains two stages: 1) calibration stage; 2) execution stage (also called the *close-loop stage*). In the close-loop mode, the system controls the external effector by means of the control signal $\mathbf{u}(t)$ in the real-time on the basis of the recordings of the brain electrical activity $\mathbf{s}(t)$. To create a control

model used at the second stage, the first stage, namely, calibration is applied. During the calibration stage, the BCI system receives the signal $\mathbf{s}(t)$ from the brain as well as information about the effector's state $\mathbf{y}(t)$ (the dotted arrow on the figure). This data is used for the BCI system parameterization, and identification of the model, representing the relation between the brain's signals and effector's state. Once all parameters of the model have been defined, the system can be switched to the next stage. In general, the feedback signal $\mathbf{y}(t)$ is not available for the control block at the second stage. At the same time, the brain has access to the feedback information (e.g., video, audio, tactile, etc.) on both stages.

Among different BCI systems, the movement-related BCIs are of great importance. Last decades several approaches were developed to face the problem of movement-related signal decoding. Promising results were obtained both in animal (Chapin et al., 1999; Wessberg et al., 2000) and in human (Leuthardt et al., 2004; Wolpaw et al., 2002) studies. To record neural activity, systems with scalp electrodes for *electroencephalography* (EEG) (Wolpaw et al., 1991; Birbaumer et al., 1999; Kubler et al., 1999), surface electrodes against the dura mater (epidural electrodes) (Rouse and Moran, 2009; Torres Valderrama et al., 2010), directly against the cortex (subdural electrodes) (Leuthardt et al., 2004), or microelectrode array for deep brain recording of single units (Chapin et al., 1999) have been used. Neural activity was analyzed using a variety of methods. Mainly the approaches were aiming at classifying of event-related electrical patterns (for instance, the case of the 'event – non-event' classification). The detection of patterns allows triggering the effector, e.g., a cursor on a screen (Wolpaw et al., 1991), a motorized device (Chapin et al., 1999; Wessberg et al., 2000), etc. Experiments were carried out either in rodents (Chapin et al., 1999; Jensen et Rouse, 2006), or in non-human primates (Wessberg et al., 2000; Serruya et al., 2002), or in human patients (Scherer et al., 2003; Leuthardt et al., 2006). Nevertheless, an effective solution of the problem still does not exist. Moreover, a set of simplifications is applied in these BCIs. For instance, the duration of the experiments usually does not exceed several minutes (for instance, see Müller-Putz et al., 2010). In addition, the subject is being given a cue to start performing a task (Wolpaw et al., 2002). And finally, experiment conditions significantly differ from the natural environment (motion

restriction, absence of significant external disturbances, and so on. See Zhao et al., 2008).

Thus, bringing a BCI out of the laboratory to real-life clinical application represents a challenging task. The development of a BCI system that meets the requirements of real applications is the main goal of the CLINATEC/LETI/CEA BCI project. The project includes development and application of implants as well as data acquisition, transferring and processing systems (Figure 1.2). The purpose of the presented work, in the frame of preclinical studying of the BCI project in Clinatec, consists in the development and implementation of methods and algorithms for the signal processing block of the BCI system. Namely, the study is dedicated to the major problems of the clinical applications of BCI: *fully autonomous functioning* of the system during *long term* in the *real-life environment*.

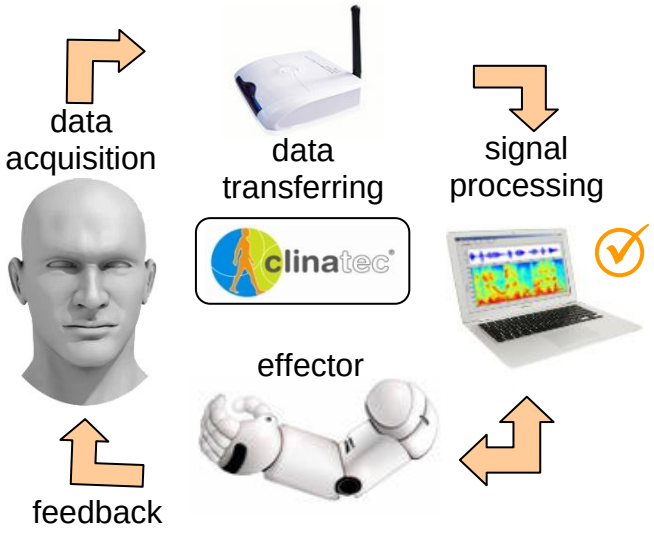

Figure 1.2 CLINATEC/LETI/CEA project: real-life human BCI.

1.2 BCI for real-life applications

BCI system for real-life clinical applications must meet a set of requirements, such as health safety, real-life conditions reliability, as well as long-term stability. The main requirements imposed on the system can be summarized as follows:

- Safety

  As the system is interacting with the brain, the safety questions, connected with risks of surgical implantation, infection, etc., have a paramount value. The use of deeper brain layers for data acquisition by means of invasive techniques increases risks for health such as brain damage or infection. Besides the risks of surgery intervention, errors in signal interpretation by the BCI system could be dangerous for the user's health and life (e.g., spontaneous motion of wheelchair and so on).

- Signal quality

  The question of the signal's quality is of great importance for BCI systems. Broad frequency band, good spatial resolution, as well as the signal-to-noise ratio of the recordings allow increase system reliability (e.g., decreasing the number of the false activations, etc.).

- System's long-time stability

  The BCI system should operate for a long period (years). Over this time, the signal can change significantly due to numerous reasons, such as the brain plasticity,

degradation of electrodes, etc. All these effects have to be properly treated by the system.

- Natural environment

  As opposed to the laboratory conditions when a subject is concentrated on the task, when external noises are minimized, and computational resources could be rather considerable, it is almost impossible to provide these circumstances in real-life environment. Thereby, the system must be robust enough to guarantee functionality of the BCI in presence of noise and spurious signals.

- Real-time conditions

  Operation in real-time is essential for the majority of BCI applications. This requirement imposes a significant restriction on both algorithms and computational facilities. Examples of real-time BCIs could be found in Chapin et al. (1999); Schalk et al. (2008); Zhao et al. (2008).

- Usability

  The system should work without everyday recalibration or this calibration should be acceptably simple and easy (for example, it takes less than several minutes per day and could be made in any conditions).

- Autonomy

  The system should be used without any external cue, whenever the user intends. Thus, the informative component should be detected on the background of the arbitrary user's activity.

- Equipment complexity and expensiveness

  Technical problems of the functional BCI system mainly consist in hardware, which should be suitable for the everyday usage in rather rigorous conditions. The BCI system must be fast enough to process all necessary information without significant time delays. From the other hand, it must be small and lightweight not to burden a person using it. Also, the questions of the long-term power supply as well as everyday stable work guarantee are of great importance. Moreover, mass production imposes limitations on the complexity and the cost of the system.

Every block of the BCI system must be optimized according to the points mentioned above. The choice of the method for the brain signals recording is a compromise of the patient's safety, signal's quality, reliability and applicability in

natural environment. The methods of data acquisition and treatment must be fast and reliable to function properly in the real-time mode in real-life conditions, etc.

In the following section, the methods and approaches of the signal recording, as well as neurological phenomena and BCI control paradigms, are compared and evaluated according to the requirements of BCI systems.

1.2.1 Neural activity recording

During the last decades, several methods were developed to measure the brain's activity. The methods range from recording activity of single neurons to large-scale brain processes analysis.

A neuron is an electrically excitable cell, which processes and transmits information by means of electrical and chemical signals. The neuron consists in a cell body (soma), an axon, and dendrites (Figure 1.3). Different neurons are connected to each other to form a neural network. In this network, signals are sent from the axon of one neuron to dendrites of others by means of electrochemical pulse (*action potential*).

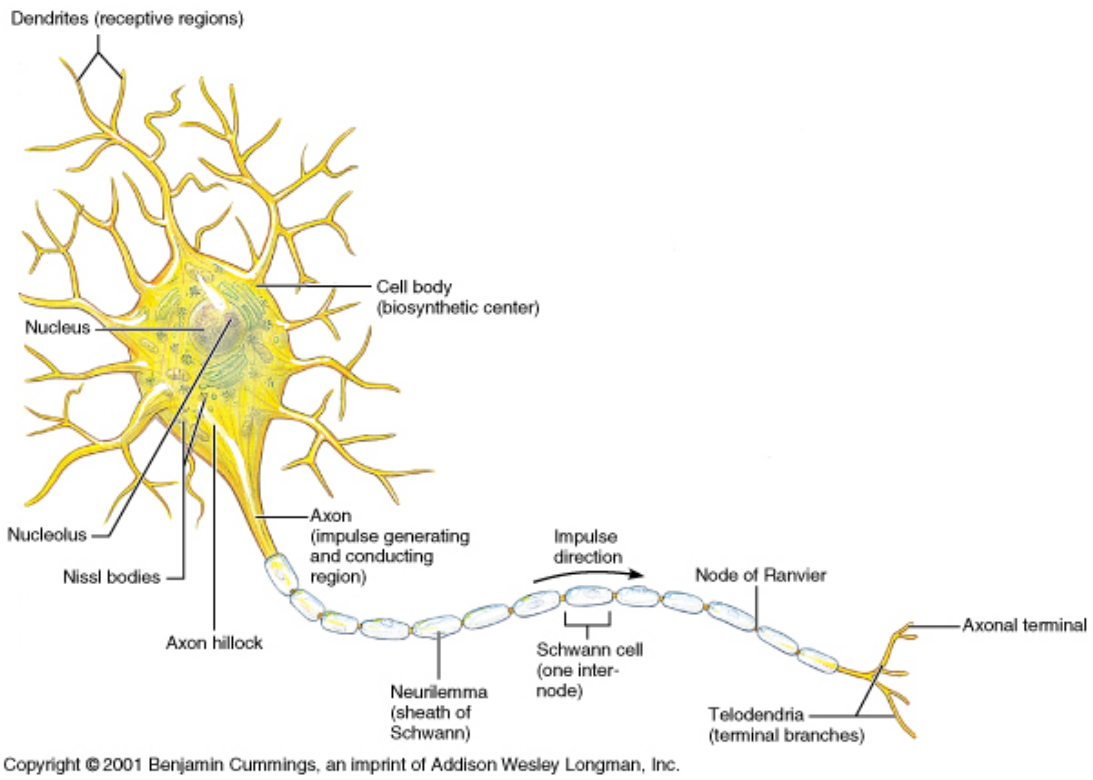

Figure 1.3 Basic structure of a neuron.

In current BCIs there are different techniques for recording of the brain's neural activity: non-invasive, partially invasive and invasive.

Non-invasive methods

The non-invasive methods are widespread due to their usability and safety. At the same time, the non-invasive signal recording systems produce poor signal resolution because the skull dampens signals, dispersing and blurring the electromagnetic waves created by the neurons.

- **Electroencephalography**

Electroencephalography (EEG) is one of the most widespread non-invasive techniques for neural activity recording firstly proposed by H. Berger in 1929. Now it is widely used as a diagnostic tool in the clinical purpose (Birbaumer et al., 1999; Wolpaw et al., 2003; Blankertz et al., 2006). Popularity of the EEG could be explained also by the simplicity of the measurements process and cheapness of the necessary equipment. EEG measurements are made by means of special electrodes with radius about 5 mm. Electrodes are mounted in the cap which can be easily worn on a subject practically without any preliminary preparation. The EEG approach consists in measurement of the small potentials (up to 100 μV) between the electrodes placed in the different locations on the scalp. Figure 1.4 shows the electrode placement in the EEG cap according to the 10-20 international system and demonstrates an example of the recorded signals.

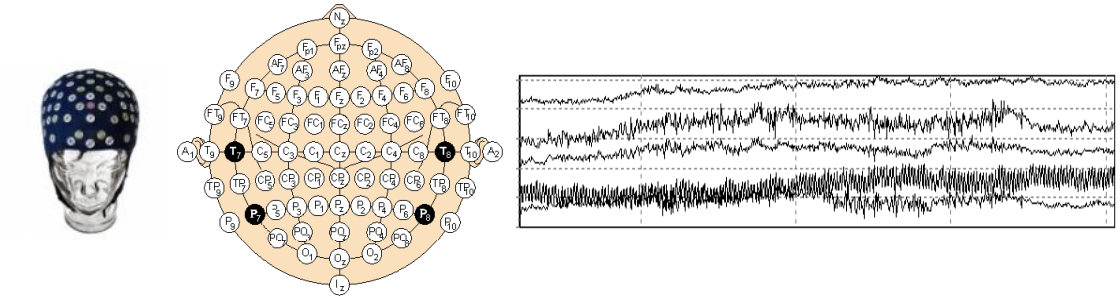

Figure 1.4 EEG cap with 10-20 international system of electrode placement on the scalp and general view of the recorded signals.

On the physical level, the signal measured with the EEG is caused by the electrical activity at pyramidal neurons in the cerebral cortex (Martin, 1991). Summation of the currents from neurons can be detected in the EEG (Sanei and Chambers, 2007). For the reason of the tissue barrier, which is between the neurons and

the electrodes, it is practically impossible to register a low-energy brain activity, as well as the frequencies higher than 100 Hz. The low limit of the EEG spectrum is 0.1 Hz. Generally the range from 0.3 Hz to 70 Hz is used in practice (Nazarpour, 2008). Moreover, artifacts from eye blinks, movements and other muscle activities complicate the analysis of EEG data. The amplitude of the artifacts sometimes significantly exceeds the amplitude of the signal of interest. Hence, often artifact filtering must be applied to the signal before data will be used.

Spectrum of the EEG is divided on several bands, which have special names (Nazarpour, 2008):

- Delta (δ): ≤ 3.5 Hz
- Theta (θ): 4-7 Hz
- Alpha (α): 8-13 Hz. An amplitude ranges generally from 20 to 100 μV, mostly below 50 μV. The α rhythm has usually the sinusoidal waveform, it is rhythmic and regular.
- Rolandic Mu (μ): 8-13 Hz. It has the same frequency band as the α rhythm but it is registered in the central and the parietal areas (see Figure 1.4). The μ rhythm has a sharp negative peak and rounded positive phase.
- Beta (β): 14-35 Hz. Located mostly in the frontocentral regions.
- Gamma (γ): ≥ 35 Hz.

The EEG is one of the most used techniques for measuring the brain electrical activity due to its safety. It is applied, for example, on research of epilepsy, sleep stages, as well as numerous clinical tasks (Acar et al., 2007). Unfortunately, shortcomings of the EEG approach significantly restrict its application in BCI systems. For instance, position of the electrodes is unstable, which force frequent system recalibration. Furthermore, training of subjects is relatively long, and some subjects can never achieve appropriate results (Blankertz et al., 2010). In addition, EEG data frequency range 0.3–70 Hz (Nazarpour, 2008) is rather low, whereas a spatial resolution (several centimeters) is too rough due to presence of the skull (Srinivasan, 1999), which dampens the signals. Moreover, as far as the recording's amplitude is rather low (≤100 μV), a significant influence of artifacts on the informative data occurs

(Hoffmann, 2007). This leads to impossibility of realization of BCIs with many degrees of freedom. Thus, while currently the EEG is widely used for the BCI systems, its shortcomings lead for searching other methods for more reliable recording of neural activity.

- **Magnetoencephalogram**

Magnetoencephalogram (MEG) is a non-invasive method for measuring the intensity of a magnetic field (Figure 1.5). In comparison with the EEG, the MEG data are less disturbed by scalp. Thus, magnetoencephalography allows obtain a signal with better spatial resolution. In addition, the MEG is more sensitive to superficial cortical activity. It is useful for studying of the neocortical epilepsy. Signal-to-noise ratio coefficient for MEG is better than the EEG one. The difference is especially significant for the high frequency band (γ-band) (Kübler and Müller, 2007).

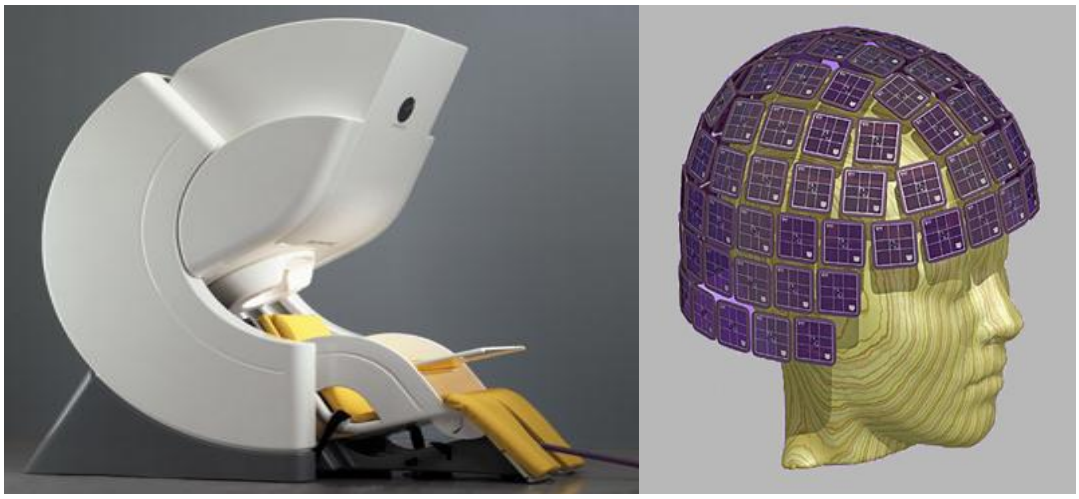

Figure 1.5 The UUMSI MEG system from Elekta Neuromag Oy in Helsinki (Finland).

The method cannot be used in real-life applications due to huge size of devices for magnetic field measurements. Moreover, the price of the equipment is very high that leads to impossibility of the MEG systems mass-production. However, some example of MEG-BCIs can be found in Hoffmann (2007); Kauhanen et al. (2006); Lal et al. (2005); Blankertz et al. (2006); Buch et al. (2008).

- **Functional Magnetic Resonance Imaging**

Functional Magnetic Resonance Imaging (fMRI) is a non-invasive method for measuring the *blood oxygen level dependent* (BOLD) signal. This method is based on

the fact that local concentration of deoxygenated hemoglobin in the brain depends on the neural activity in the cells. In comparison to the EEG, the fMRI allows detecting activity not only in the cortex but also in deeper structures of the brain (Weiskopf et al., 2004). Figure 1.6 demonstrates an example of the fMRI scanner as well as examples of the obtained brain images.

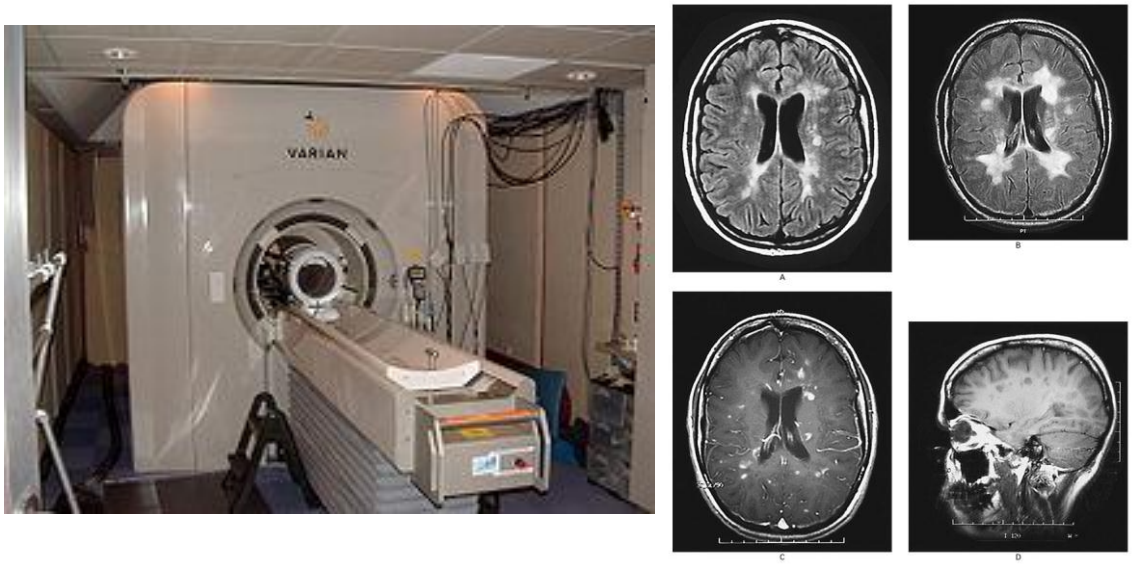

Figure 1.6 Berkeley's 4T fMRI scanner and brain images example.

Unfortunately, the fMRI like the MEG measurements are very complex and expensive. They can only be made in laboratory conditions and cannot be easily applied in practical BCI systems.

- **Near Infrared Spectroscopy**

Near Infrared Spectroscopy (NIRS) is a non-invasive method for the measurement of hemodynamic activity. NIPS uses near-infrared light, which depends on the amount of oxygenated and deoxygenated hemoglobin in the blood. In contrast to the fMRI, the NIRS is significantly cheaper and requires less technical effort, whereas the spatial resolution is almost the same. The method was applied for the motor imagery and the movement tasks in BCI systems (Coyle et al., 2004; Sitaram et al., 2005; Sitaram et al., 2007). However, an application of NIRS in the real-life BCI is still not possible nowadays due to the complexity of the equipment.

Thus, among the non-invasive methods, mainly EEG is applied in the practical BCI systems. The other methods are mainly used for fundamental researches and medical purposes.

Invasive methods

- **Microelectrode arrays**

Microelectrode arrays (MEA) are a technique for registering activity of small groups of neurons (*local field potentials*) or even a single neuron (single-unit action potentials). To record signals, needle-like electrodes are placed into the brain cortex during surgery. First arrays were implanted by C. Thomas in 1972 (for more information, see Pine, 2006). Electrode diameter ranges between 10 and 30 μm. The method allows recording of the signals with frequency up to 5000 Hz. Examples of the MEA electrodes and signal are given in Figure 1.7. On average, the microelectrode array is covering a surface up to about 50 mm^2.

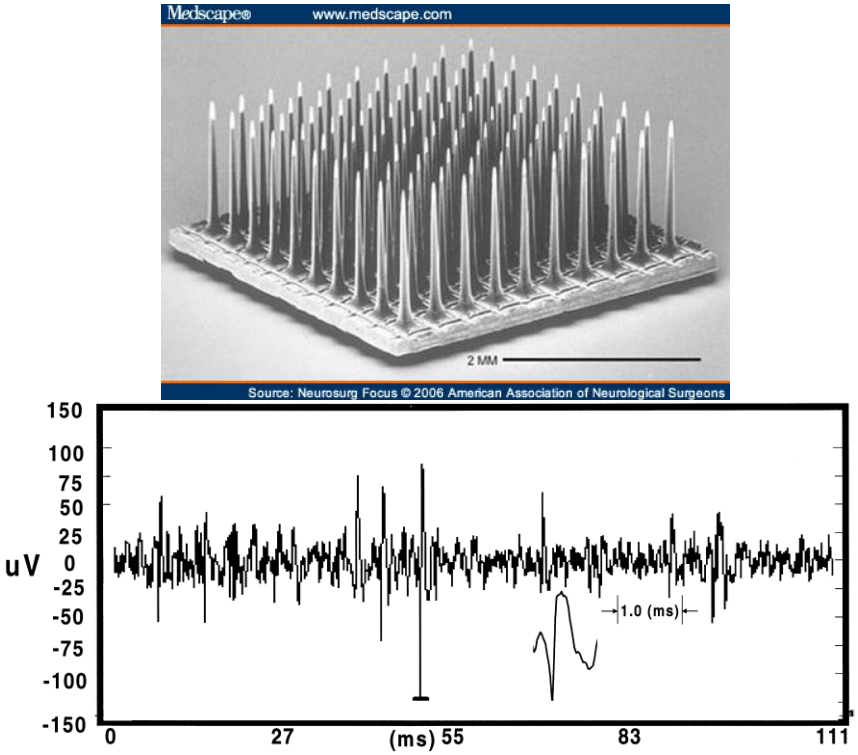

Figure 1.7 The silicon-based microelectrode array: 100 electrodes spaced at 400-micron intervals (Normann et al., 1999) and an example of the registered signal (Williams et al., 1999).

The main advantage of the MEA application in comparison with other approaches is its high spatial resolution, which gives opportunity to apply the microelectrode arrays for complex tasks, e.g., real-time 3D motion (Taylor et al., 2002) and robotic arm control (Wessberg et al., 2000; Schwartz, 2004). The action-potential firing rate allows estimation of the subject's intention of movement (Serruya et al., 2002.). For instance, 96 electrodes were implanted in a human subject suffering from tetraplegia that makes possible to the subject to operate with a prosthetic hand and to accomplish other tasks (Hochberg et al, 2006).

The main disadvantage of the MEA application is the penetration of the cortex to a depth of several millimetres (Nicolelis et al., 2003). Moreover, quality of the registering signals decays with individual neuron death over time (Biran et al., 2005). Besides that, the astrocytes and microglia adhere to the device begin to accumulate. They form a sheath surrounding the array. This both increases the space between the electrode probes and insulates the electrodes. In addition, this increases a measurement impedance. Thus, the data can be recorded only for several months. On the other hand, recent results (Hochberg et al., 2006) demonstrate that a stable recording over a long time (year) is possible but at the expense of losing the signal at numbers of electrodes (about 20 of 150 remained functioning).

The MEA is a remarkable tool for the brain signal registration. Unfortunately, due to the safety questions the microelectrode arrays are used mainly in animals. Additional information about the method can be found in the review of Lebedev and Nicolelis (2006).

Thus, the method provides a high quality signal with significant spatial and frequency resolutions, as well as a good signal-to-noise ratio. As far as response time of the system is small enough (Taylor, 2007), whereas the system is compact and relatively cheap, it could be effectively applied for BCI tasks. On the other hand, the method is rather dangerous due to presence of significant risks of the brain damages. In addition, the system long-term stability supply is rather laborious. Moreover, only a small region of the brain could be observed.

Partially-invasive methods

- **Electrocorticography**

The *Electrocorticography* (ECoG) uses an electrode grid or strip to record the electrical activity of the brain's cortex. The ECoG was pioneered by W. Penfield and H. Jasper in the 1950's and now it is considerably widespread (Huggins et al., 1999; Leuthardt et al., 2004; Felton et al., 2007; Schalk et al., 2008). Like the EEG, the ECoG is based on measuring of the potential activity in the cortical pyramidal neurons. The difference is that the electrodes are placed immediately on the cortex surface. Thus, the electrical signals must not be conducted through the skull, where their potentials diminish due to the low conductivity of the bone. It leads to increasing of the spatial resolution of the ECoG in comparison with the EEG (tens of millimeters versus centimeters), broader frequency bandwidth (above 300 Hz), and higher signal's amplitude (50-100 μV). In addition, the ECoG recordings are less influenced by artifacts (Leuthardt et al., 2004).

Generally, the ECoG experiments are made in the context of the long-term clinical applications which require continuous monitoring and processing of the neural activity for epileptic patients. For reach the cortex, a craniotomy must be made for a patient, i.e., a part of the skull must be removed to expose the brain surface (see Figure 1.8). This procedure is performed under general or local anesthesia. The electrodes arrays can be placed either on the surface of the dura mater (epidural) or beneath the dura mater (subdural). The electrodes are attached to a flexible frame. Standard spacing between the electrodes in the frame is 1 cm; diameter of the electrodes is up to 5 mm (Schuh and Drury, 1996). The electrodes are designed in the way to eliminate any injury of the brain during its movements.

Figure 1.9 demonstrates electrodes arrangement in the brain. Figure 1.10 represents the cortex with the electrodes placed for the subdural extraoperative ECoG.

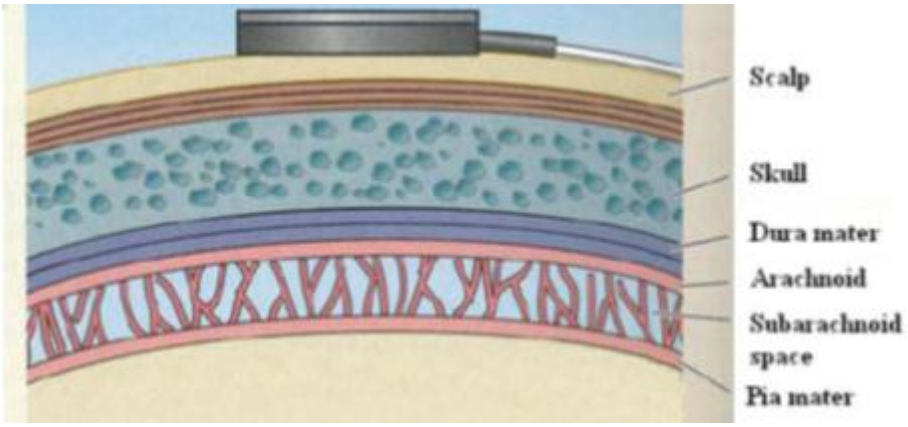

Figure 1.8 Tissue layers covering the cortex.

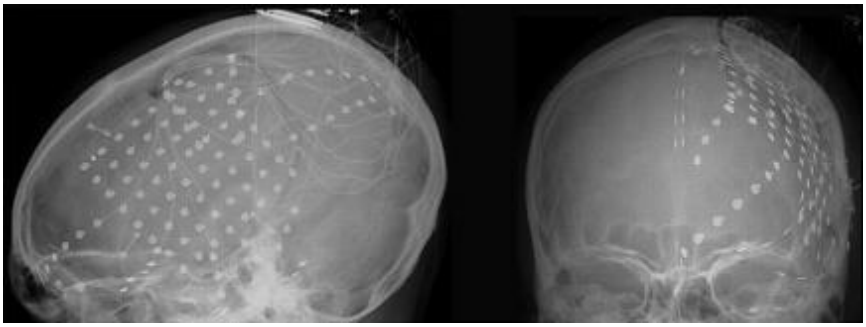

Figure 1.9 Spatial electrode arrangement shown through MRI image (Kutsy, 1999).

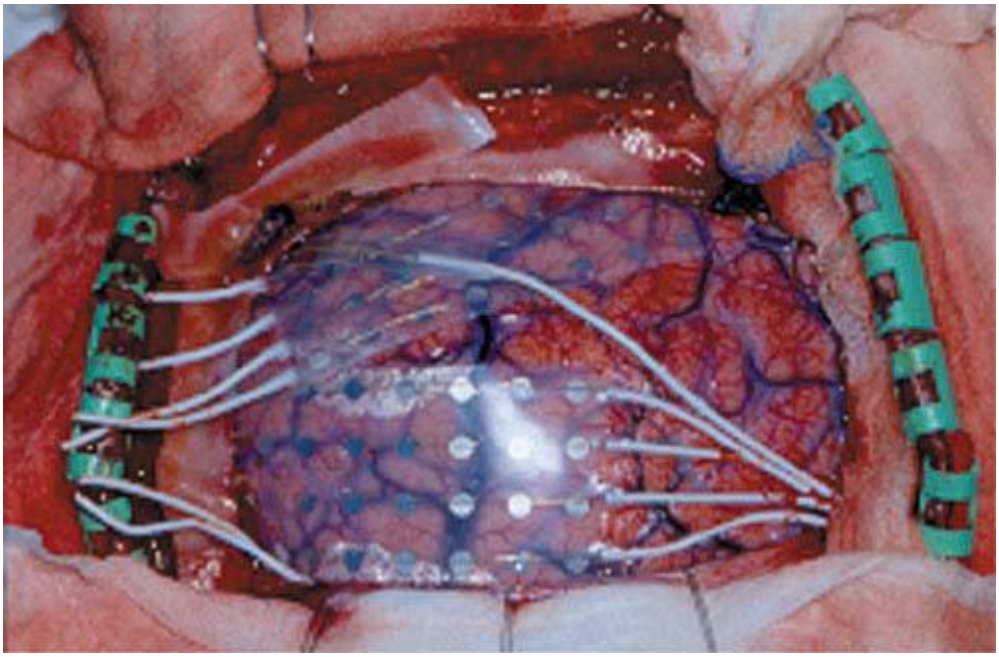

Figure 1.10 Exposed cerebral cortex and electrode placement for the subdural ECoG (Miller et al., 2007).

Besides the clinical applications, the ECoG is considered as promising technique for the BCI systems. It does not penetrate the brain significantly, but has higher spatial and frequency resolutions in comparison with the EEG. Moreover, it has better signal-to-noise ratio, which is the very important characteristic for BCIs. The BCI experiments are carried out in subjects with implanted matrix of electrodes. Therefore, the tests cannot be made in healthy humans. Commonly, ECoG is used to localize seizure activity in patients with epilepsy before they undergo surgery (Kübler and Müller, 2007). The ECoG experiments have demonstrated that the users can learn to control the ECoG signals through motor imagery (Graimann et al., 2004; Hill et al., 2006), speech imagery (Leuthardt et el., 2006), auditory imagery (Wilson et al., 2006), as well as mental calculation (Ramsey et al., 2006). Most of ECoG BCI studies were carried out offline and performed open-loop analysis of data (Kübler and Müller, 2007; Huggins et al., 1999; Levine et al., 1999). Scherer et al. (2003) reported accuracy between 85 and 91 percent for offline classification of data recorded during self-paced middle finger extension. Hill et al. (2006) achieved accuracies around 75 percent for motor imagery tasks. However, only in few studies close-loop was realized and feedback of ECoG was provided to the participants (Felton et al., 2007; Leuthardt et al., 2004; Leuthardt et al., 2006; Wilson et al., 2006). Birbaumer (2006) implanted electrodes on the cortex of a completely paralyzed patient to classify motor imagery signals. Unfortunately, classification results were at the chance level. Nevertheless, the carried out experiments demonstrated that even more than one year after implantation 50 percent of the electrodes provide stable signal recording. The promising results from new designs of the ECoG matrix electrodes for the long-term registration of the neuronal activity were reported recently (Charvet et al., 2011; Ejserholm et al., 2011; Hirata et al., 2010). Contrary to the EEG-, ECoG-based BCIs avoid problems with muscular artifacts (Sutter, 1992; Zaveri et al., 1992), offers better localization of origin of the signals (Salanova et al., 1993), a wider frequency range (Leuthardt et al., 2004), and higher signal-to-noise ratio (Margalit et al., 2003). In addition, there is a benefit of shorter training times for the ECoG-based BCIs (Leuthardt et al., 2004; Wolpaw et al., 2002; Huggins et al., 2007).

Thus, the quality of the signal recorded with the ECoG is appropriate to be used in the BCI project. The frequency band and the spatial resolution, as well as the signal-

to-noise ratio surpass correspondent parameters of the non-invasive methods. The risk of implantation is not as high as for the microelectrode arrays. Furthermore, since the recording does not depend on a single or several neurons, the system provides better long-term stability than the invasive methods. Response time and size of the system allows its utilization in the real environment in the real-time mode. Hence, as a tradeoff between registered signal quality and safety, the ECoG was selected as the method for the brain activity recording in the Clinatec project.

Nanostructured electrodes for the ECoG recordings

One perspective direction for ECoG recording is application of nanostructured electrodes. The advantages of this approach (in particular, carbon nanotubes (CNT) on the surface of the electrodes) were reported recently by Sauter-Starace et al. (2009). The electrodes covered by CNT allow registration of higher current in comparison to the electrodes without CNT. Their signal-to-noise ratio is better (Keefer et al., 2008).

A set of experiments were carried out to compare the quality of the signals recorded using CNT covered electrodes vs. usual electrodes. The electrode implant was made on a 2 cm x 4 cm flexible 120 μm-thick polyimide support. A copper layer, contacting pads and the electrode contact areas were included in polyimide layers. 2 x 16 electrodes made of silicon with a surface of 1 mm² covered of titanium nitride (TiN) and CNTs were disposed in the contact areas (see photos in Figure 1.11).

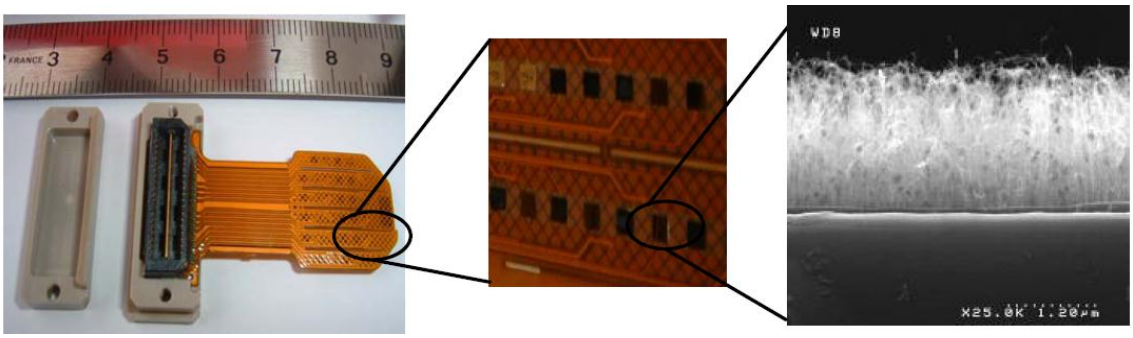

Figure 1.11 CNT electrodes on the flexible structure (Sauter-Starace et al., 2009).

Carbon nanotubes were grown by chemical vapor deposition (CVD) technique, according to a protocol reported in details by Dijon et al., 2004. A 3 nm-thick layer of nickel was deposed on TiN samples to catalyze the carbon nanotubes synthesis. Analysis of obtained CNTs (see Figure 1.12 (A)) demonstrated that CNTs layer were around 2 μm thick with CNTs diameters ranging between 20 and 40 nm.

To test the electrodes in vivo they were implanted in a monkey. Brain activity was recorded weekly over one year. Typical registered ECoG signal is demonstrated in Figure 1.12 (B). It could be seen from the figure that signals from CNTs electrodes are richer than ones from TiN electrodes. This observation was consistent over the course of the study, irrespective to variation in amplitude (between 50 μV and 200 μV) and in the level of noise (between 20 μV and 50 μV). Power spectra represented in Figure 1.13 also demonstrates advantage of CNTs electrodes.

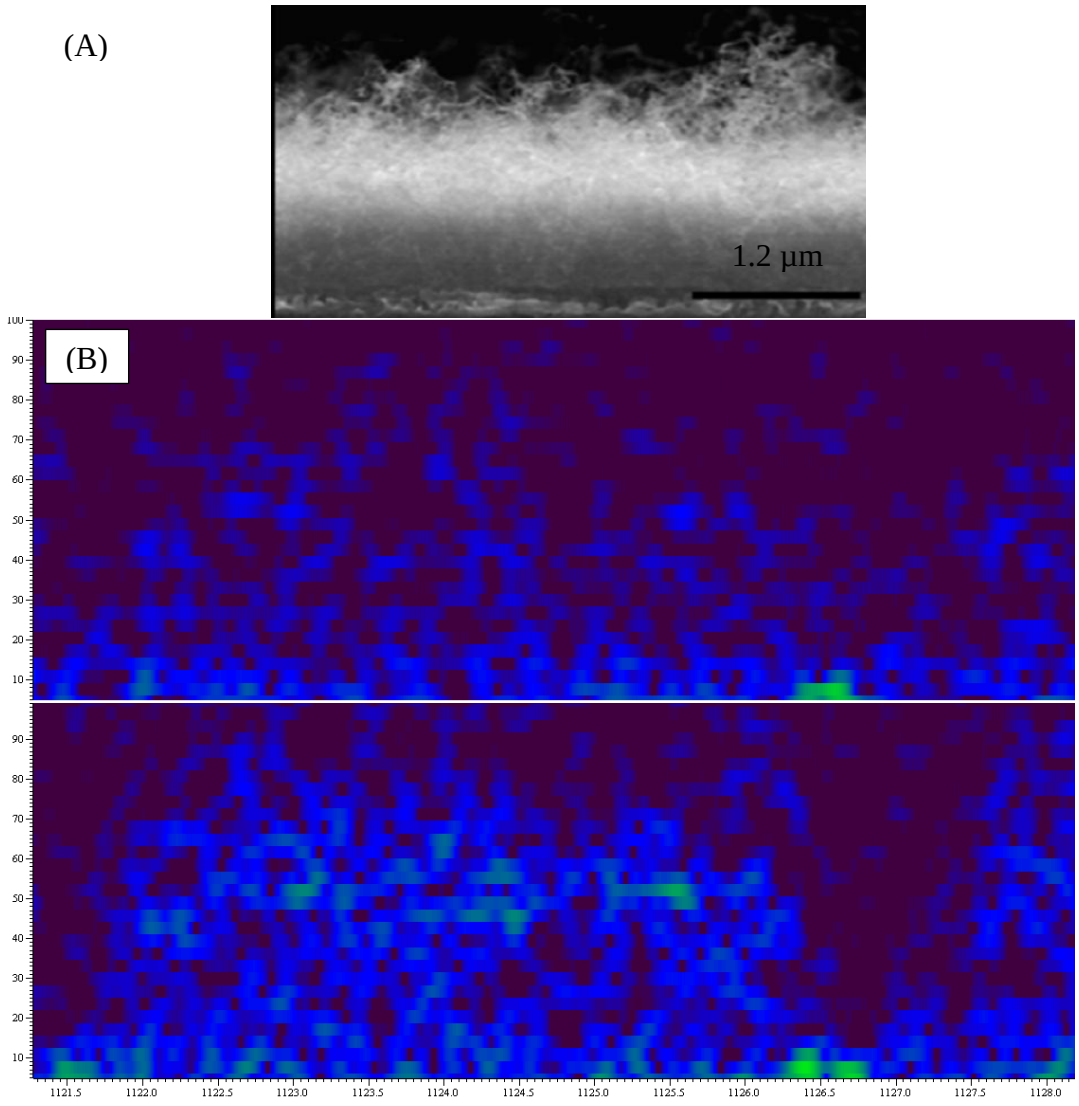

Figure 1.12 (A) Scanning electronic microscopy (SEM) picture of carbon nanotubes (Sauter-Starace et al., 2009). **(B)** Typical recorded ECoG signal (Sauter-Starace et al., 2009). Top and bottom pictures are sonograms of TiN and CNT electrodes, respectively (here, x: time in seconds, y: frequency in Hz).

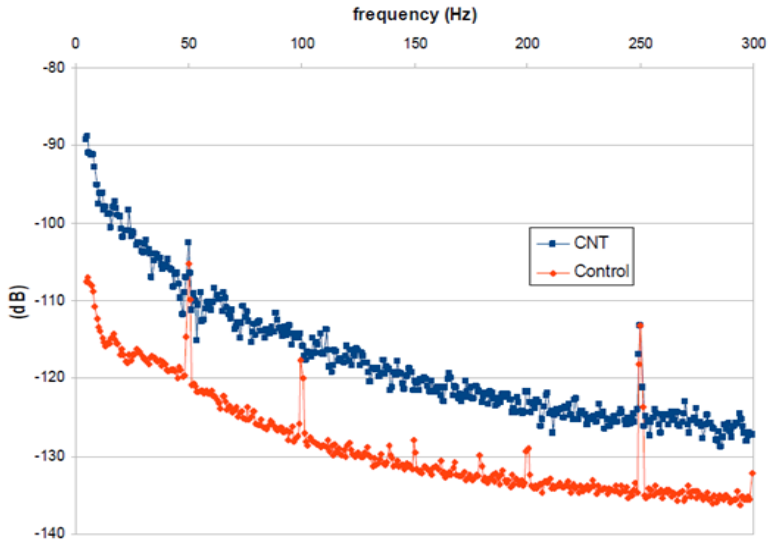

Figure 1.13 Comparison of the power spectra of ECoG. CNTs electrode is blue, TiN (control) electrode is red (Sauter-Starace et al., 2009).

The experiments demonstrated that during several months the implanted CNT electrodes were able to record neuronal signals. Any epileptic activity, which could be linked to brain irritation, was not observed. Nanostructuration with CNTs is reported as an effective way to improve signal quality by increasing the signal-to-noise ratio in ECoG.

Thus, carbon nanotubes improve quality of the registration of the brain neuron electrical activity. Long-term biocompatibility tests were carried out in vitro by Sauter-Starace et al (2006). At the same time, some long-term experiments in vivo were made to date. However, to have more reliable results the additional tests are required.

1.2.2 Electrophysiological sources of control in BCIs

Numerous neurological phenomena could be used to control the BCI system. These phenomena are characterized by voltage amplitude, latency, frequency and space distribution. Some of them are described below.

- *Slow Cortical Potentials (SCPs)*

  SCPs are the slow potential changes generated by a user. Usually they have a non-movement source and reflect the changes in the cortical polarization lasting from 300 ms up to several seconds (Birbaumer et al., 1990). The CSP based BCI systems require users to achieve voluntary regulation of its brain

activity. They typically use the traditional S1-S2 paradigm which was proposed in the work of Walter et al. (1964). A negative SCP shift seen after a warning stimulus (S1) two to ten seconds before an imperative stimulus (S2) that requires participants to perform a task (e.g., a button press or cursor movement). An example of the SCP application for the BCI system could be found in Birbaumer et al. (1999). In this work patients suffering from amyotrophic lateral sclerosis (ALS) use the BCI to control a spelling device.

- *P300*

  Auditory, visual and somatosensory stimuli can evoke a positive peak at about 300 ms after the stimulus is received. This positive signal deflection is called P300 (Squires et al., 1976). The P300 amplitude varies. It depends on discriminability of standard and target stimuli, overall probability of the target stimuli, and the preceding stimulus sequence, as well as the electrodes position (Squires et al., 1976). Mostly, the P300 is observed in the central and the parietal regions.

  Farwell and Donchin had shown that the event-related potential could be used to select items displayed on a computer monitor (Farwell and Donchin, 1988). From users it was required that they were able to focus attention and gaze on the target letter for a considerable amount of time.

  For more information see Nieuwenhuis et al. (2005).

- *Visual Evoked Potentials (VEP) and Steady-State Visual Evoked Potentials (SSVEP)*

  Small changes in the brain signal generated in response to a visual stimulus are called VEPs. Their characteristics depend on the type of the visual stimulus. If a visual stimulus is presented repetitively at a rate of 5-6 Hz or greater a response is termed SSVEP. Its amplitude and phase depend on such stimulus parameters as repetition rate and contrast. Like the P300, the SSVEP requires attention and intact gaze but no user training.

  The SSVEP-BCI was used, for instance, in the work of Wang et al. (2006). Three targets with different flickering frequencies were presented on a monitor. On general, up to 90% classification accuracy is reported. However, a shortcoming of all SSVEP approaches to the BCI control is their dependence on

intact gaze. Hence, the method is unsuitable for patients with restricted eye movement. Moreover, it cannot be used in long-term experiments.

Description of the BCI systems using the SSVEP could be found in Herrmann (2001), Gao et al. (2003), and Lalor et al. (2005).

- *Sensorimotor rhythms (SMR)*

  The sensorimotor rhythm is brain wave rhythm which appears over the sensorimotor cortex. SMRs are characterized by a decrease (desynchronizing) and increase (synchronizing) of the signal's energy in specific bands (Pfurtscheller and da Silva, 1999). Moreover, SMRs are evoked by the real movements and by the motor imagery (when no actual movement is required, which is the only possible solution for paralyzed patients). Voluntary modulation of SMR could be achieved by the subject after the first session (Blankertz et al., 2006). Moreover, it was demonstrated by Kübler et al., 2005 that SMR regulation is possible even in the case of significant degeneration of cortical and spinal motor neurons. However, the amplitude of SMR is much higher in healthy individuals (Kübler and Müller, 2007). Some examples the BCI systems made on the base of the SMRs could be found in Blankertz et al. (2006), Pfurtscheller and Neuper (2001), Wolpaw and McFarland (2004).

The objective of our BCI system is autonomous functioning in the real-time mode. Therefore, among all described above phenomena only sensorimotor rhythms could be utilized, whereas others occur with significant time lag and/or cannot be used in the real environment without external stimulation (SCP, P300, VEP, and SSVEP).

1.2.3 Robustness and stability of BCI system

Robustness and stability are important questions for real-life application of the BCI systems. Essential restrictions are imposed on the stability of the system due to a considerable variability of the registered signal. There are three main types of the signal's variability (Tangermann, 2008):

- subject-to-subject variability
- session-to-session variability for the same subject
- trial-to-trial variability for the same session and the same subject

The detailed study of the neurological signal variability is presented in the work of Tangermann (2008). An example of the subject-to-subject variability is shown in Figure 1.14. The averaged brain patterns exhibit a great diversity for 6 subjects for the case of right and left hand finger typing.

Characteristics of the signals, recorded from the same subject during the different sessions, i.e., in different days, could also vary (session-to-session variability) Figure 1.15. While in the short-time (one day) experiments this type of instability could be neglected, in the long-term experiments it should be taking into account.

The trial-to-trial variability represents instability of the registered signal from one trial to another for the same subject and during the same session. Figure 1.16 demonstrates an example of this type of variability.

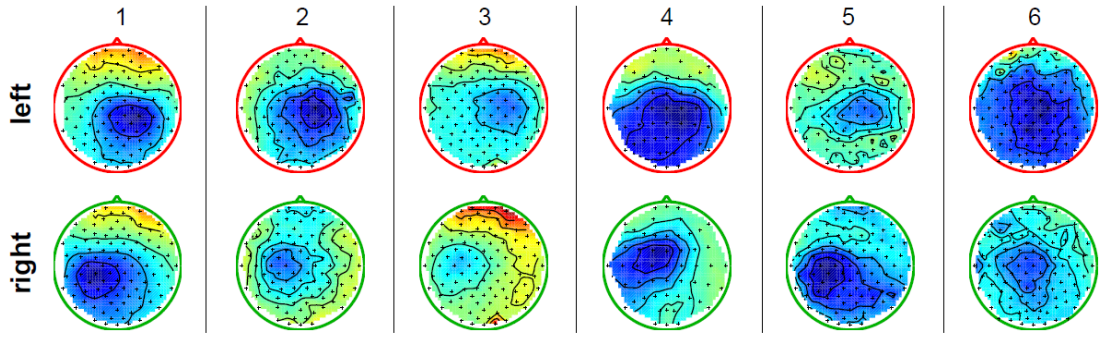

Figure 1.14 Example of the subject-to-subject variability (Tangermann, 2008). Represents the trial-averaged brain patterns made for the 6 subjects for the case of right- and left-hand finger typing.

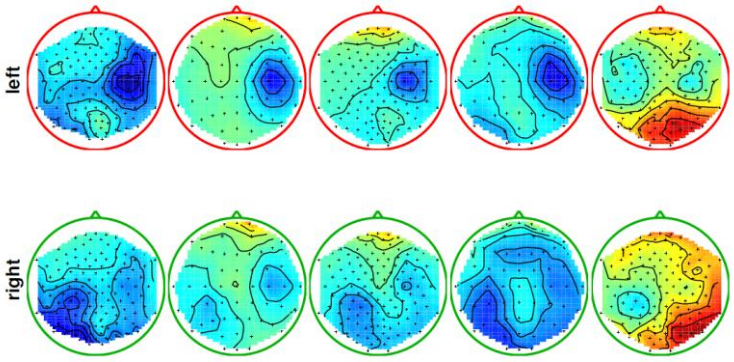

Figure 1.15 Example of the session-to-session variability (Tangermann, 2008). Represents an average across 140 trials made for the same subject (different sessions are shown on the different columns) for the case of right- and left-hand finger typing.

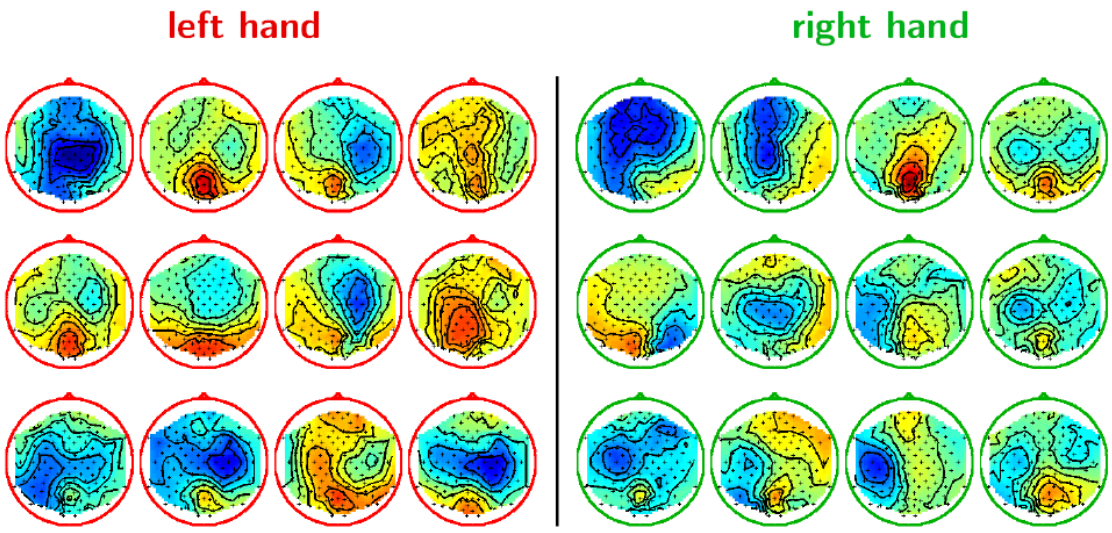

Figure 1.16 Example of the trial-to-trial variability (Tangermann, 2008). It represents the alpha-band power for twelve 3.5 seconds-length trials for the case of right- and left-hand finger typing.

The functional BCI system has to operate in the real-life. Therefore, the signal's variability should be taken into consideration. To take into account the effects of the session-to-session instability of the signal, all currently existing BCI systems require an everyday pre-calibration of the parameters. Moreover, online adaptation of the algorithm could allow adjustment of the method to the data variations during one session. Taking into account different types of the signal's instability, generally, the algorithm used in the BCI system must be sufficiently robust, i.e., its decisions should be made on the base of characteristics which are minimally exposed to the variability of the data recordings.

1.2.4 BCI Control paradigms. Self-paced BCI

BCI systems can be classified according to the way of the *no control* (NC) periods supporting, when the BCI system is expected to remain in the neutral state, in the other words, to be idle. Moreover, in the real BCI, the NC states happen more frequently than the *intentional control* (IC) states. Currently, several control paradigms are used in the design of the BCI systems. According to Mason and Birch (2005), they can be classified as:

- self-paced (asynchronous)

- system-paced
- synchronized (cue-based)

A scheme of the different control paradigms is represented in Figure 1.17.

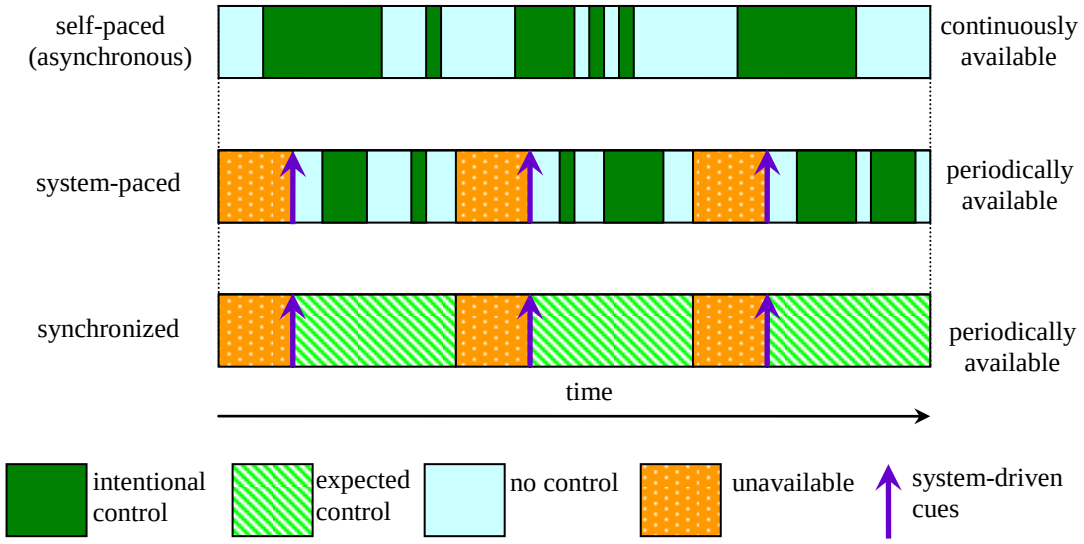

Figure 1.17 Schematic representation of the different control paradigms (Mason and Birch, 2005).

Only the self-paced and the system-paced paradigms support the NC states. On the other hand, the synchronized and the system paced control strategies use external cues for driving an interaction between a subject and the BCI system. As a result, users are supposed to generate commands only during specific periods. The signals outside the predefined time windows are ignored. However, in the real-life environment this restriction seems to be very burdensome as well as causes user frustration and fatigue. As opposed to the cue-paced systems, no stimulus is used by the self-paced BCIs, which is continuously available. Users control them at their own intention. However, the performance of the self-paced BCIs reported in the articles is still not suitable for practical application, in particular, because of high level of false system activations (Fatourechi et al., 2008). It also causes frustration of the user and limits the applications of the system.

Different control paradigms are characterized by various neurological phenomena on which the BCI system is based. The self-paced paradigm can utilize the sensorimotor rhythm phenomenon so long as its nature is not connected with presence of any predefined cue. At the same time, the system-paced BCI can be based on the

slow cortical potentials, whereas the synchronized BCIs could work with the P300 or the visual evoked potentials (the steady-state visual evoked potentials) since all these phenomena are strictly connected with some predefined event which can be used as cue.

The self-paced control approach is appropriate for the real BCI systems, as far as it provides more freedom and control flexibility to users. However, it is more difficult to realize such systems, since there are many different types of brain's activity (e.g., sleeping, eating, different mental tasks, etc.). As a result, lots of reported BCIs are synchronous, for instance, see Wolpaw et al. (2002). Nevertheless, recently several groups have pursued design of the self-paced BCI systems (e.g., Leeb et al. (2007); Scherer et al. (2008); Fatourechi et al. (2008); Müller-Putz et al. (2010); Qian et al. (2010)).

To concentrate on the main problem, in this work we are considering the simplest case of the BCI system, i.e., the case of the binary self-paced BCI, also known as the brain-switch BCI. The brain-switch BCI was introduced by Mason and Birch (2000), as the system intended to detect only one brain state (brain pattern) in the ongoing brain activity and does not provide any output when the user does not intend to communicate (Pfurtscheller et al., 2010a). Mason and Birch system was able to distinguish voluntary motor-related potentials from ongoing EEG activity in asynchronous mode, by analyzing a low-frequency band. Last time single channel EEG-based brain switch were realized by Pfurtscheller et al., (2005); Pfurtscheller and Solis-Escalante, (2009); Solis-Escalante et al., (2010). A multiple-electrode brain-switch system was created recently by Barachant et al., (2011). Except EEG, NIRS (Coyle et al., 2007) also can be used for BCI. Pfurtscheller et al., (2004) demonstrated an ECoG-based brain switch BCI system. Whereas all previously mentioned BCIs were exploiting SMR, brain switches were also realized on the basis of SSVEP (Cheng et al., 2002). Taking into account advantages of the brain-switch approach, it was chosen for the first version of our BCI system. At the same time, applied algorithms should be easily generalizing to the case of multi-states and continuous-states BCI system, which is the future goal of our project.

In the next section criteria for the self-paced BCI performance evaluation and comparative analysis are discussed.

1.2.5 Evaluation of the self-paced BCI performance

Criteria

The binary BCIs classify incoming data epochs on two types, based on presence or absence of some specific activity. Epochs are classified as ‘event’ or ‘non-event’.

For evaluation of the binary BCI, several approaches are used, for example, the error rate or the classification accuracy (Blankertz et al., 2004), mutual information and the information transfer rate (Kronegg et al., 2005), the receiver-operator-characteristics (ROC) and the area-under-curve (AUC) (Lal et al., 2005), the correlation coefficient (Gao et al., 2003) and the mean square error (MSE) (Wu et al., 2006). However, these criteria were mainly applied for the synchronous type of the BCI control. The specificity of the self-paced systems consists in significantly different probabilities of ‘event’ and ‘non-event’ classes. Generally, long periods of the NC are interspersed with brief instances of the IC. It means that the NC states occur with a much higher probability than the IC ones, which contradicts the underlying assumption of equal a priori probability for a variety of the traditional performance estimation methods.

Another important characteristic of the BCI performance, which should be taken into account, is the decision rate, i.e., a number of decisions produced per unit of time. It could be identical to the sampling rate of the input signal (e.g., Huggins et al., 1999) or could be decimated to some extent (e.g., Kübler et al., 2005). It is clear that the greater the decision rate is the more the absolute number of errors could be made if the probability of the error remains the same. It should be taken into account during interpretation of the performance properties.

Following characteristics are used to evaluate classification:

- True Positive (TP), which is equal to the number of correctly detected ‘events’;
- True Negative (TN), which is equal to the number of correctly detected ‘non-events’;
- False Positive (FP), which corresponds to the number of ‘non-event’ situations detected as ‘events’;
- False Negative (FN), which is the number of the missed ‘events’.

These four characteristics are summarized in the confusion matrix (Table 1.1).

Table 1.1 Confusion matrix contains information about the number of the correct detections, event missings, false activations, and correct rejections.

	'Event'	**'Non-event'**
'Event' detection	TP (correct detection)	FP (false activation)
'Event' rejection	FN (event missing)	TN (correct rejection)

Figure 1.18 gives a graphical representation of these characteristics for a case when the decision rate coincides with the signal sampling rate.

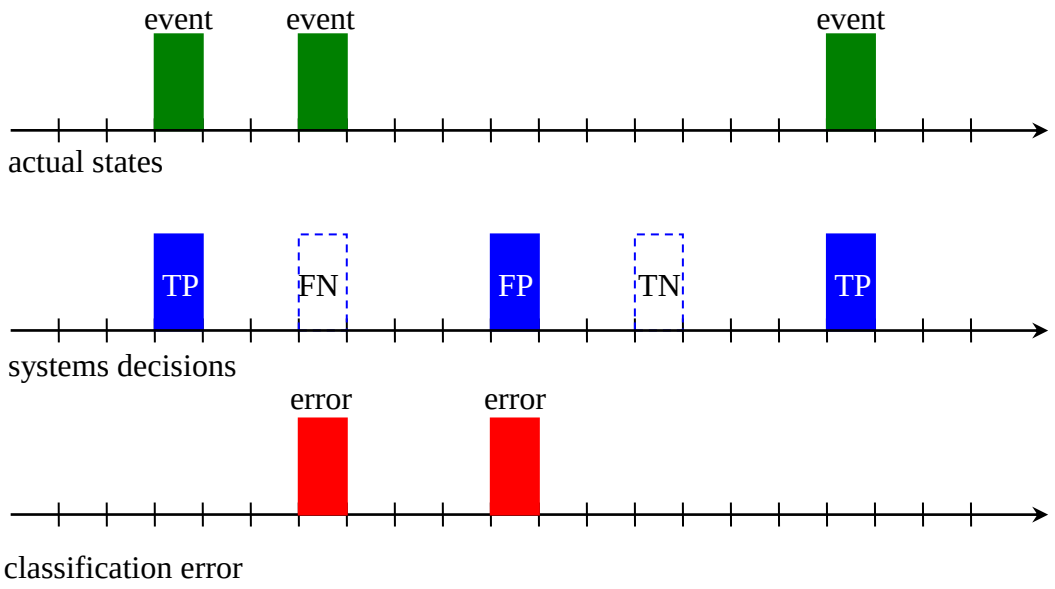

Figure 1.18 Graphical representation of the decision characteristics.

The ECoG signal could not be perfectly synchronized with the moments of the events (for instance, due to the response delay of the system). Thus, online quality evaluation of the self-paced (binary) BCI system should be made. In the current study it is realized in the following way. The event is considered to be detected correctly (TP) if the system has detected it during the certain (predefined) time interval from the real event (see Figure 1.19 (A)). After every generated detection the system is blocked and cannot produce any other detection during some predefined period of time (see Figure 1.19 (B)). It is made to prevent the multiple system actuation for the same event.

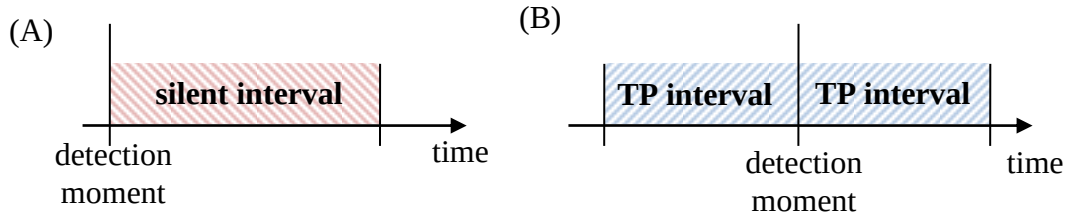

Figure 1.19 (A) After each detection the system cannot generate other detections during the predefined time interval. **(B)** Detection is treated as True Positive (TP) if it is made during the certain time interval from the real event.

On the basis of TP, TN, FP, and FN values, criteria for the system performance evaluation are introduced. Among the most frequently used are (Huggins et al., 1999; Rijsbergen, 1979; Schlögl et al., 2007; Fatourechi et al., 2007):

- Accuracy of Classification: $ACC = \frac{TP+TN}{TP+TN+FP+FN}$
- Error Rate: $ERR = 1 - ACC$
- True Positive Rate (Hit Rate, Sensitivity): $TPR = \frac{TP}{TP+FN}$
- False Positive Rate: $FPR = \frac{FP}{FP+\mathrm{TN}}$
- False Detection Rate: $FDR = \frac{FP}{TP+FP}$
- False Activation per Minute
- Positive Predictive Value (Precision): $PPV = \frac{TP}{TP+FP} = 1 - FDR$
- Specificity: $Sc = \frac{TN}{TN+FP}$
- Harmonic Mean of Precision ($F_1 - measure$): $F_1 = 2\frac{TPR \cdot PPV}{TPR+PPV} = \frac{2 \cdot TP}{2 \cdot TP+FN+FP}$
- Hit/False-difference: $HFdiff = TPR - FDR = TPR - PPV + 1 = \frac{TP}{TP+FN} - \frac{FP}{TP+FP}$

The accuracy of classification (ACC) is a commonly used evaluation criterion in the BCI research. It shows the percentage of correctly classified samples. However, being efficient for the cue-paced BCIs, ACC, as well as the error rate ERR, fail to characterize the performance of a self-paced BCI due to highly unbalanced classes

(Schlögl et al., 2007). The True Positive Rate (TPR), and the False Positives Rate (FPR) are widely used in the self-paced BCIs. They show the relative amount of successfully detected events and of false activations. Let us note that FPR is influenced by the decision rate and the ratio of the classes. This complicates the BCI systems comparison. That is why additional criteria characterizing false activations of the self-paced BCIs were proposed: the number of the False Activation per Minute (Mason and Birch, 2000), the Positive Predictive Value (PPV) (Müller-Putz et al., 2010), etc. While TPR shows the percentage of successfully detected events, PPV corresponds to the percentage of correct detection.

The confusion matrix and/or a set of criteria, for instance, TPR and FPR, allow characterizing the BCI performance. However, simultaneous comparison of several criteria is not convenient for the calibration purpose. Since standard ACC fails to evaluate the performance of the self-paced BCIs, numerous attempts were made to introduce a single metrics: the weighted ACC (Zhu and Yao, 2004), HF-difference (Huggins et al., 1999), F1-criterion (Rijsbergen, 1979), TPR at a fixed false positive rate (Borisoff et al., 2004), ratio TPR/FPR (Fatourechi et al., 2007) and others. In our study, the average value of TPR and PPV (referred as the Overall Performance, $OP = (TPR + PPV)/2$) was chosen for the calibration purpose. We apply it to optimize the detection threshold in the BCI system ('event' – 'non-event'). OP gives a single value for the BCI performance, balancing the False Positives and False Negatives types of errors. The choice of the criterion for the BCI system calibration reflects the goals and priorities of the given BCI study.

Thus, by means of the Overall Performance characteristic, the quality of the BCI system could be measured by a single value.

Whereas the BCI system has to work in the real-time conditions, the response time (RT) of the system generally is of great importance. It could be characterized by the probability of the different values of the response lags. An example of the response-time histogram is represented in Figure 1.20.

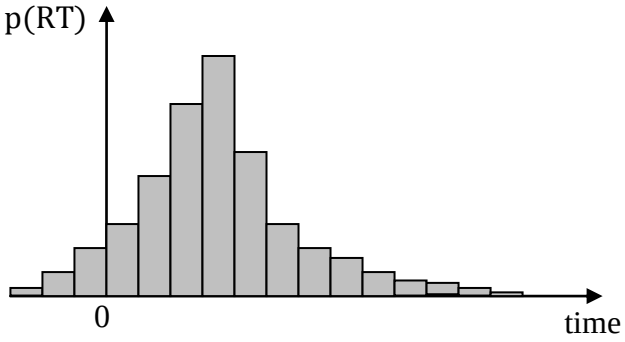

Figure 1.20 An example of the Response-Time Histogram: represents the probability of the different response delays.

Comparison

One of the most significant problems of the self-paced BCI systems is high value of the False Positive Rate (Mason and Birch, 2000), i.e., the number of the false activation during the NC periods. While the current BCI realizations have rather good detection rate (the True Positive Rate or, in other words, the rate of the correct detections of the IC states, see Fatourechi et al., 2008), the False Positive Rate is still too high for practical applications. For example, if the system makes a decision every 100 ms and has the FP rate equal to 1%, it will generate an error on average every 10 seconds that is unacceptable in the numbers of the real applications due to user's frustration or other reasons.

Low performance of the current self-paced BCI systems (Müller-Putz et al., 2010; Mason and Birch, 2000; Yom-Tov and Inbar, 2003; Townsend et al., 2004; Bashashati et al., 2006; Fatourechi et al., 2006; Bashashati et al., 2007a; Fatourechi et al., 2008) is the main restriction for the practical BCI applications. Most of the reported BCIs achieve less than 60% of the control intention detection (TP) with several false positives (FP) activations per minute (Mason and Birch, 2000; Yom-Tov and Inbar, 2003; Townsend et al., 2004; Bashashati et al., 2006; Bashashati et al., 2007a). The BCI system reported by Fatourechi et al. (2008) achieved 56% of correct detection, with 0.7 FP per minute: this was obtained by offline analysis of short recordings of only several (2-5) minutes length and of non-control periods when a subject was resting. These results were obtained from the scalp EEG recordings in highly restricted conditions and

are not satisfactory in the context of the needs of the long-term clinical applications which require continuous monitoring and processing of the neural activity.

1.2.6 Functional BCI in natural environment

Stable functioning in the natural environment conditions is required for the applications of the BCI systems. Previous BCI experiments were carried out in the laboratory conditions, which distinguish significantly from the natural environment where subjects are not concentrated properly, can be disturbed by external noises, etc. As an approximation to the natural conditions, the BCI experiments in freely moving animals (rats and non-human primates) were carried out. The animals continue their general behavior during the experiments. The self-paced BCI system in freely moving animals gives an opportunity to avoid any external control cues, i.e., the animal could control the BCI system barely based on its own intention. Thus, the BCI experiments in freely moving animal were carried out since they model the real-environment conditions.

1.2.7 BCI cognitive tasks. Motor imagery

It is known that paralyzed patients have difficulties in operating with the BCI because of absence of sensory feedback. Nevertheless, it is still possible to use a phantom movement as a basis of the BCI system (*BCI cognitive task*). Imagination of the movements produces similar to the real-movement activity in the brain; however specificity of the patterns is smaller (Anderson et al., 1995; Gu et al., 2009; Pfurtscheller and da Silva, 1999; Beisteiner et al., 1995; Wolpaw et al., 2003). An example is given in Figure 1.21.

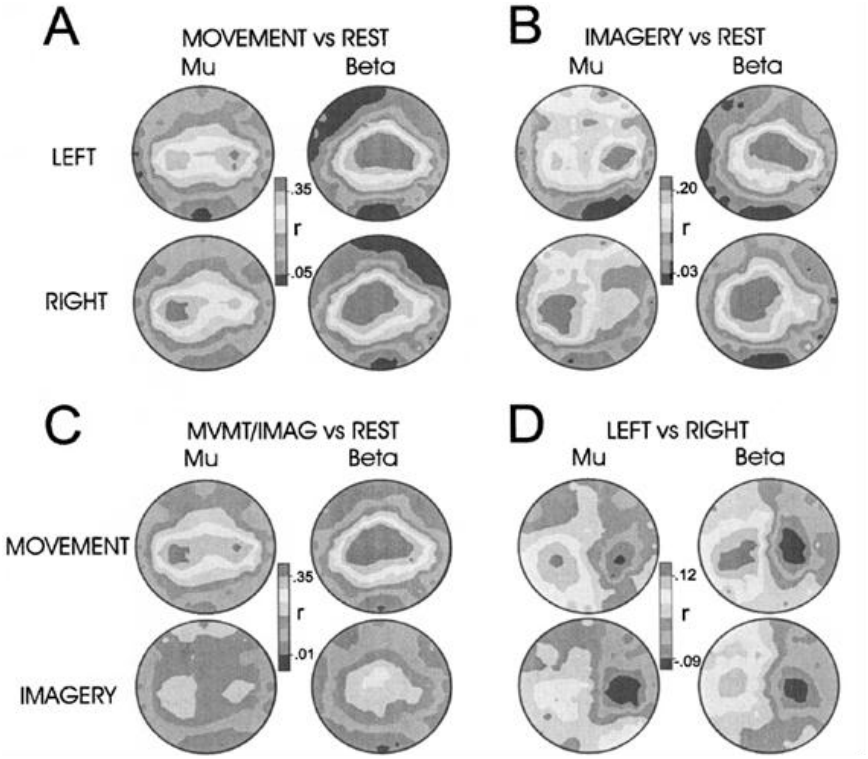

Figure 1.21 Topographies for mu- and beta-band activity. **(A)** Left-hand movement versus rest and right-hand movement versus rest. **(B)** Left-hand imagery versus rest. **(C)** Movement versus rest and imagery versus rest. **(D)** Right- versus left-hand movement and imagery. Taken from McFarland et al., (2000).

In the current study, we concentrate on the case of real motions, because of difficulty to organize the motor imagery experiments in animals. However, simplicity of the SMR responses for the real and imagery motions allows us to hope for future successful use of the developed methods for the cognitive BCI tasks.

1.2.8 Conclusions

In the frame of the preclinical study of the human BCI system in the context of the Clinatec project, the goal of this thesis is development of algorithms for the signal processing block for the *long-term experiments with ECoG-based self-paced binary BCI in freely moving animals*.

The ECoG recording of the neural electrical activity is made directly from the surface of the brain cortex. Contrary to other invasive or partially invasive methods, it guarantees high level of safety for subject. The risk of brain damage or infecting is sufficiently moderate. At the same time, the recorded electrical signals are not

conducted through the skull. Thus, high spatial and frequency resolutions of the recorded data are provided. Moreover, influence of the artifacts considerably diminishes while the informative signal's amplitude increases. The response time of this registration method is small enough to use the electrocorticography in the real-time mode. In addition, the ECoG system is rather convenient for the everyday use. It is cheap and can be mass-produced relatively easy. Therefore, it was chosen as method of the data acquisition in the Clinatec project.

The self-paced control paradigm is best suited to the experiment when no driving cues are used and a subject takes actions at its own intention. This paradigm provides to the user more freedom and command flexibility in comparison to the other control paradigms. Even taking into account difficulties in development and implementation of the self-paced methods, this approach was chosen for the project as the most promising and the best appropriate for the concept of the planned experiments.

While this work is making as the preclinical prototype of the system operating with patients under the natural conditions, it is crucial to create the system able to function adequately in the environment close to the real-life conditions. The absolute majority of the previously carried out experiments have been essentially limited to the laboratory conditions, at the same time, the whole experiments as well as each their session were relatively short. Contrary to this, our goal is the BCI system which treats properly different forms of animal's behavior, such as eating, sleeping, resting and so on during the long-term experiments. Thus, it could be taken as a satisfactory model of the human long-term real-life BCIs.

For the clinical purpose, significant number of BCI systems should continuously handle an external effector (e.g., prosthesis, wheelchair, etc.). At the same time, the discrete control can approximate the continuous control with any level of precision. We have chosen the binary BCI, which utilizes the simplest case of the discrete control. Thus, we can concentrate on other problems.

Therefore, the *long-term functional ECoG-based self-paced binary BCI in freely moving animal* unites applicability and large practical importance. The problem statement lets us focus on the most significant issues of the task of development of the preclinical prototype of the BCI system.

Chapter 2. BCI experiment setup

2.1 Installation setup for rodents

The BCI experiments are carrying out by the neurophysiology group of Clinatec. The author was participating in definition of the experiments' protocol and assisted in the experiments. The experiments are based on a simple reward-oriented task. Freely moving in the ABETT® behavioral cage (Abett II Starter kit, Campden Instruments; Lafayette Instrument Co., Leicestershire, UK) rat has an opportunity to push a pedal, mounted in a wall of the cage, see Figure 2.1. Every pushing event activates a food dispenser and the rat obtains a reward (a food pellet). The rat is trained to press the pedal without any cue or conditioning stimulus.

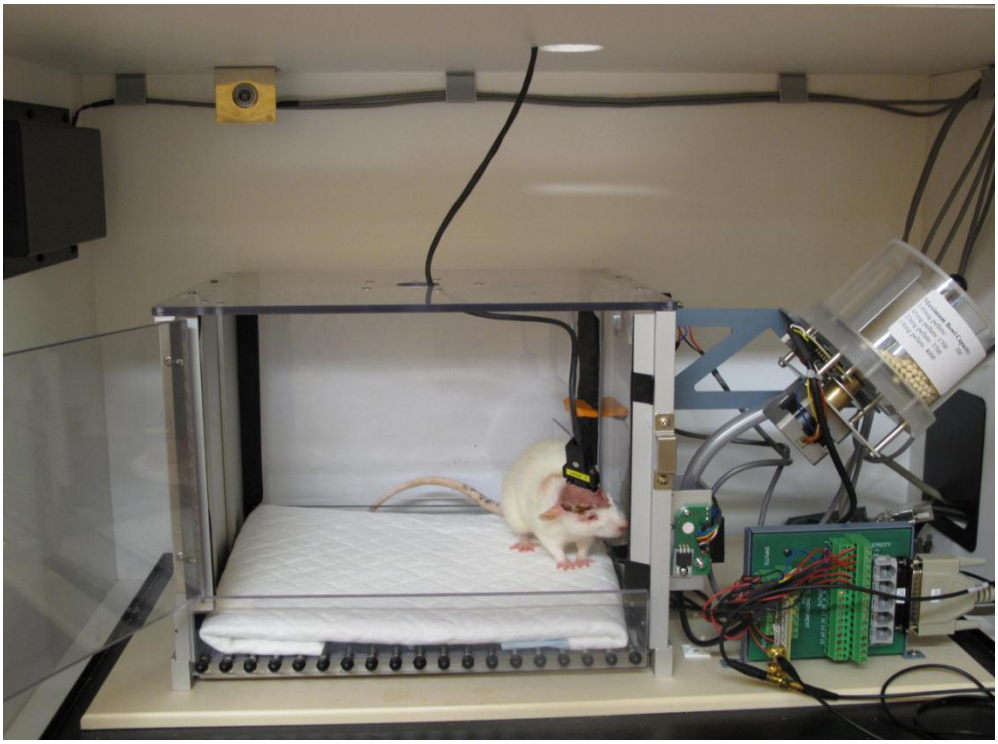

Figure 2.1 Photo of the installation for the experiments in rats (CLINATEC/LETI/CEA).

The ECoG signals from the brain of the rat were simultaneously recorded through 14 transcranial screws. Three additional electrodes were used as references, i.e., to distinguish and eliminate a common source noise. Data were continuously recorded by Micromed® system (Micromed SD64, Micromed Italy) as well as by BioMEA system (modular 256-channel Micro-Electrode Array, Biologic, France) compatible with Matlab® based signal processing. The acquisition scheme was developed and implemented by CEA/LETI/DTBS/LE2S, Grenoble, France (Charvet et al., 2011). Preliminary visual inspection of the recorded signals was made in Spike2® software (Cambridge Electronic Design Limited, UK).

Positions of the electrodes implanted in the brain of the rat are demonstrated in Figure 2.2, whereas Figure 2.3 gives an example of the recorded signals (visualization made by Spike2® software). The lines ##1-11 of Figure 2.3 represent the ECoG recordings of the rat. The specificity of the data consists in plenty of chewing artifacts, which are produced by the rat's jaw muscles. These artifacts mainly are located after pressing event and have different duration. The 12th line demonstrates the state of the pedal ('pressed' or 'released').

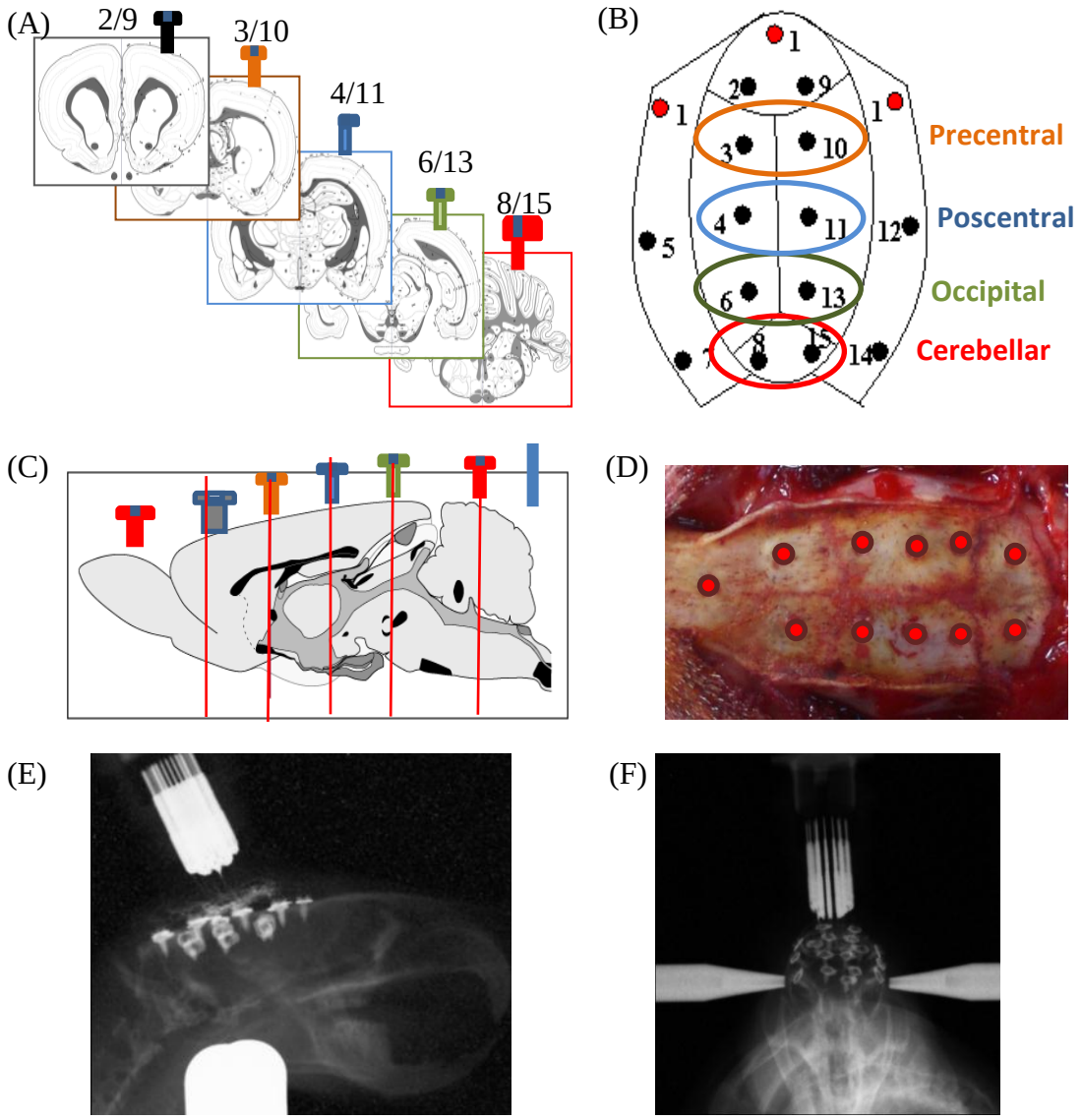

Figure 2.2 Position of the electrodes implanted in the rat's brain. Anatomical distribution of the electrodes on atlas (L. Swanson plates) coronal sections **(A)** and on sagittal view **(C)**. **(B)** from back to front, the electrodes are situated over the cerebellum (retro-lambdatic: 8, 15), the occipital (visual area: 6, 13), the postcentral (4, 12), the precentral (3, 10), and the prefrontal (2, 9) cortices. 4 additional electrodes are temporal, left and right: anterior (5,12), and posterior(7,1). **(D)** Skull distribution. **(E)** and **(F)** represent X-Ray images of the implanted electrodes (CLINATEC/LETI/CEA).

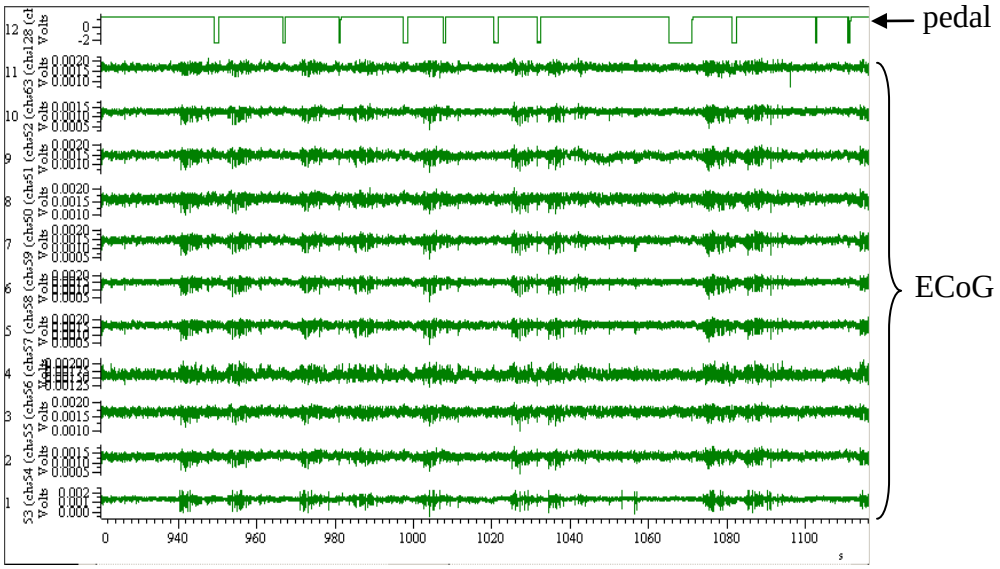

Figure 2.3 Fragment of the registered signal represented by Spike2® software (Cambridge Electronic Design Limited, UK).

2.2 Installation setup for non-human primates

The BCI experiments in monkeys are carrying out by the neurophysiology group of Clinatec (the author was participating in definition of the experiments' protocol). Data was collected from behavioral experiments in non-human primates (Rhesus Macaque) based on a simple reward-oriented task. During the experiment the monkey is sitting in a custom made primate chair minimally restrained, its neck collar hooked to the chair, see Figure 2.4. The monkey has to push a pedal which can be mounted in for different positions ('left', 'right', 'up', and 'down') on a vertical panel facing the monkey (Figure 2.5). Every correct push event activates a food dispenser and the monkey obtains a reward (fruit juice). No cue or conditioning stimulus were used. A set of ECoG recordings was collected from 32 surface electrodes chronically implanted on the surface of monkey's cortex (see Figure 2.6). The electrode matrix was produced by the Ad-Tech Medical Instrument Corporation (Racine, USA). Diameter of the electrodes is 2 mm. Figure 2.7 gives an example of the recorded signals. Information about the state of the pedal was stored simultaneously with ECoG data.

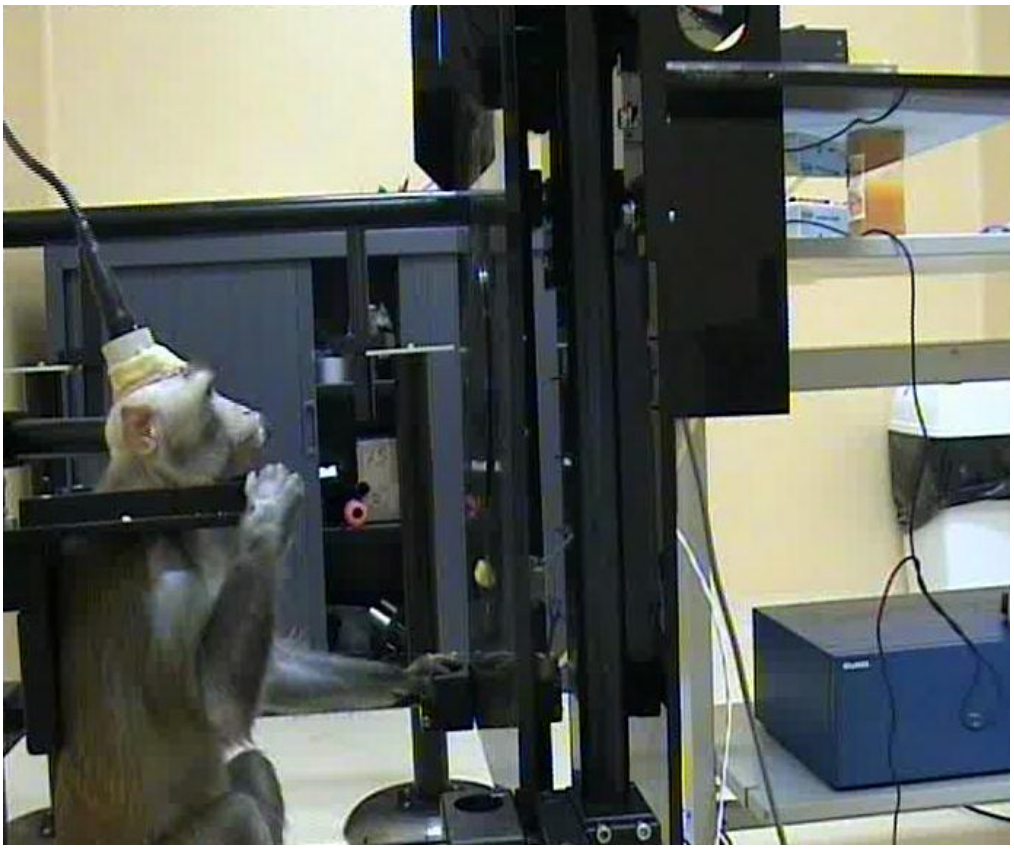

Figure 2.4 Photo of the installation for the experiments in monkeys.

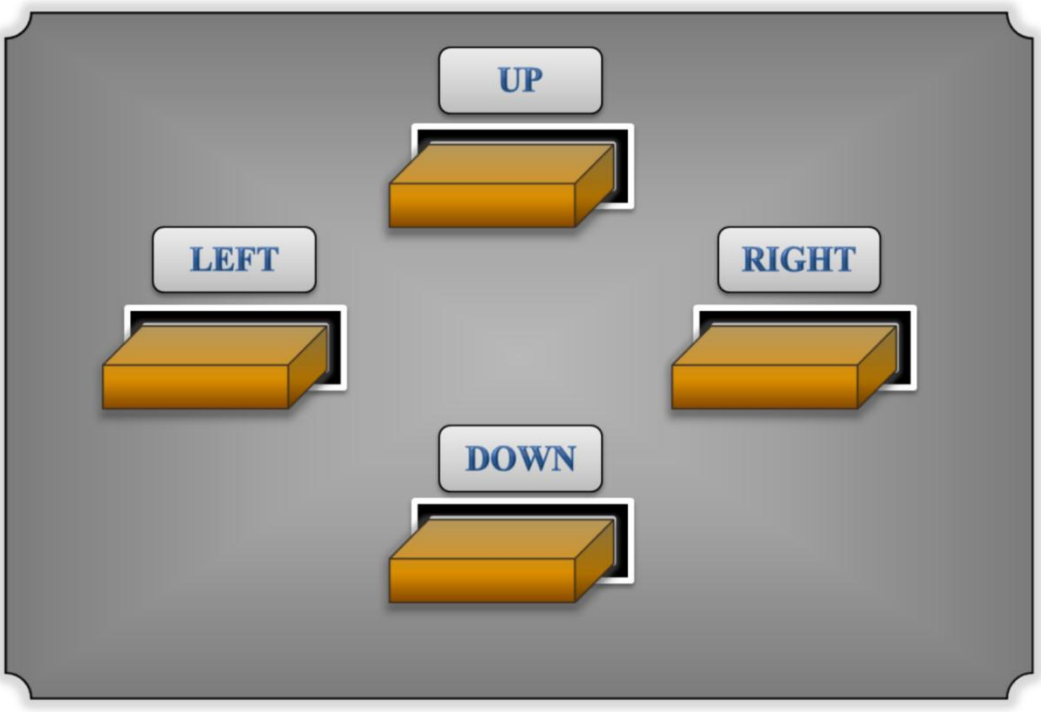

Figure 2.5 Four possible positions of the pedal on the panel (not to scale).

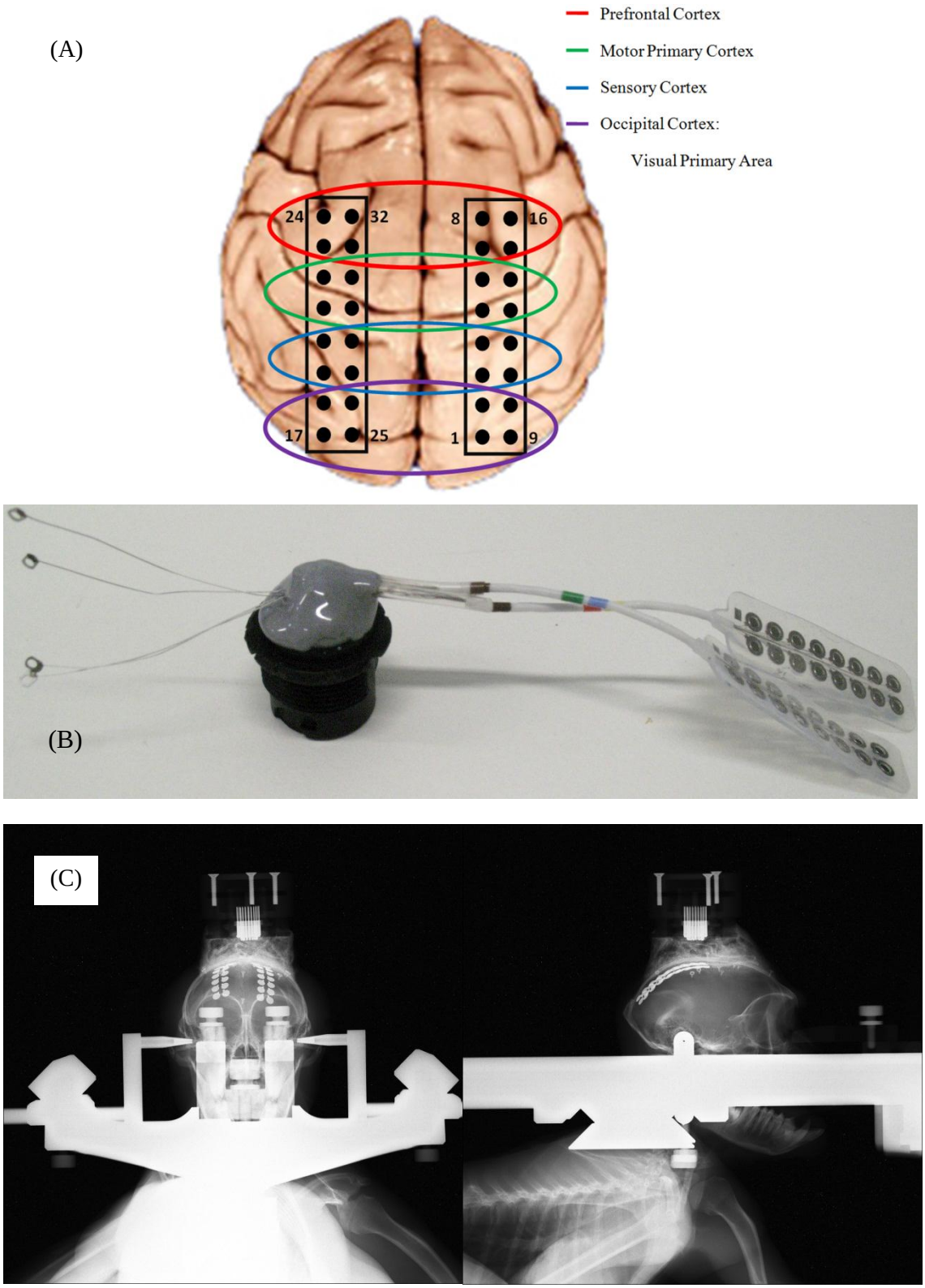

Figure 2.6 (A) Position of the electrodes implanted in the monkey's brain. **(B)** The photo of the implant. **(C)** X-Ray images of the implanted electrodes (CLINATEC/LETI/CEA).

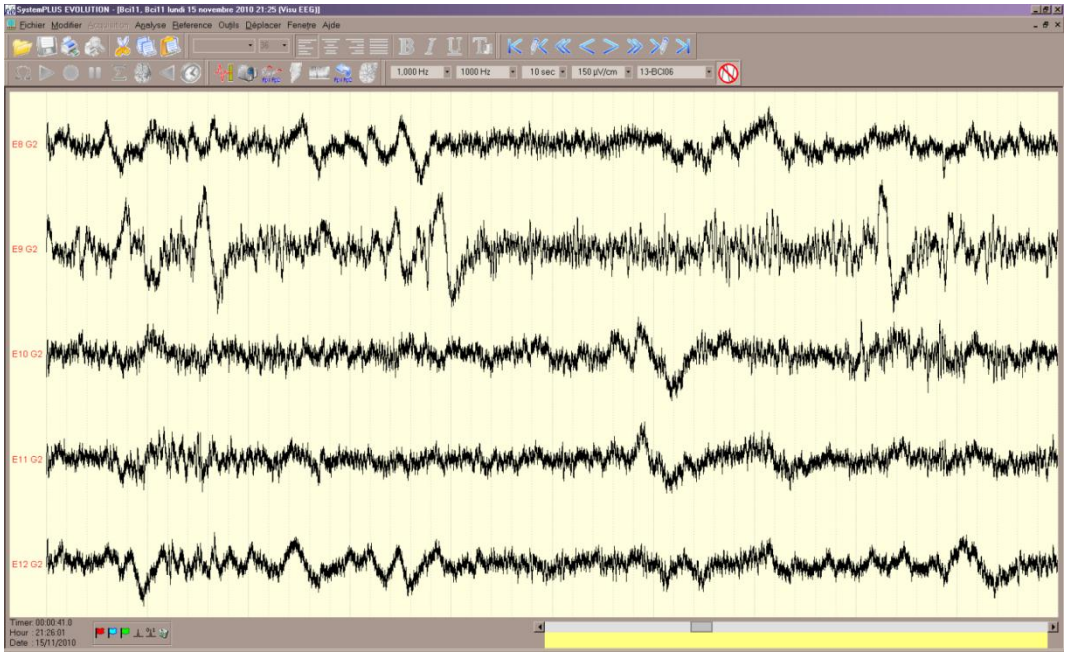

Figure 2.7 A fragment of the recorded signal represented by Micromed® SystemPlus Evolution® software (Micromed SD64, Micromed, Italy).

2.3 Experimental setup for rodents

The BCI experiment in rat consists of two stages (see Figure 2.8). The first one is the Training Stage. During this stage the freely moving rat can voluntary press the pedal without any external cues. The food dispenser is activated by pressing the pedal. Thus, after every pushing the rat obtains a reward, namely one food pellet. During the experiment session the ECoG data from the rat's brain (denoted as **S** in Figure 2.8) is recorded simultaneously with information about state of the pedal ($y(t)$ equals to 1 if the pedal is pressed at the moment t and 0, otherwise). Those experiments, which contain less than 50 pushes, are neglected. One of the others forms a training set which is used for the offline BCI system calibration. Identified model is applied for the 'pushing-event' prediction.

At the second stage, namely the Close Loop Real-Time BCI, the predictor is directly applied to the signal from the rat's brain in the real-time. On this stage, the pedal is disconnected from the reward distributor. Thus, the rat presses the pedal but it obtains a reward only in the case when the predictor detects a 'pushing-event' (whatever the position of the pedal is) and generates required control signal $u(t)$.

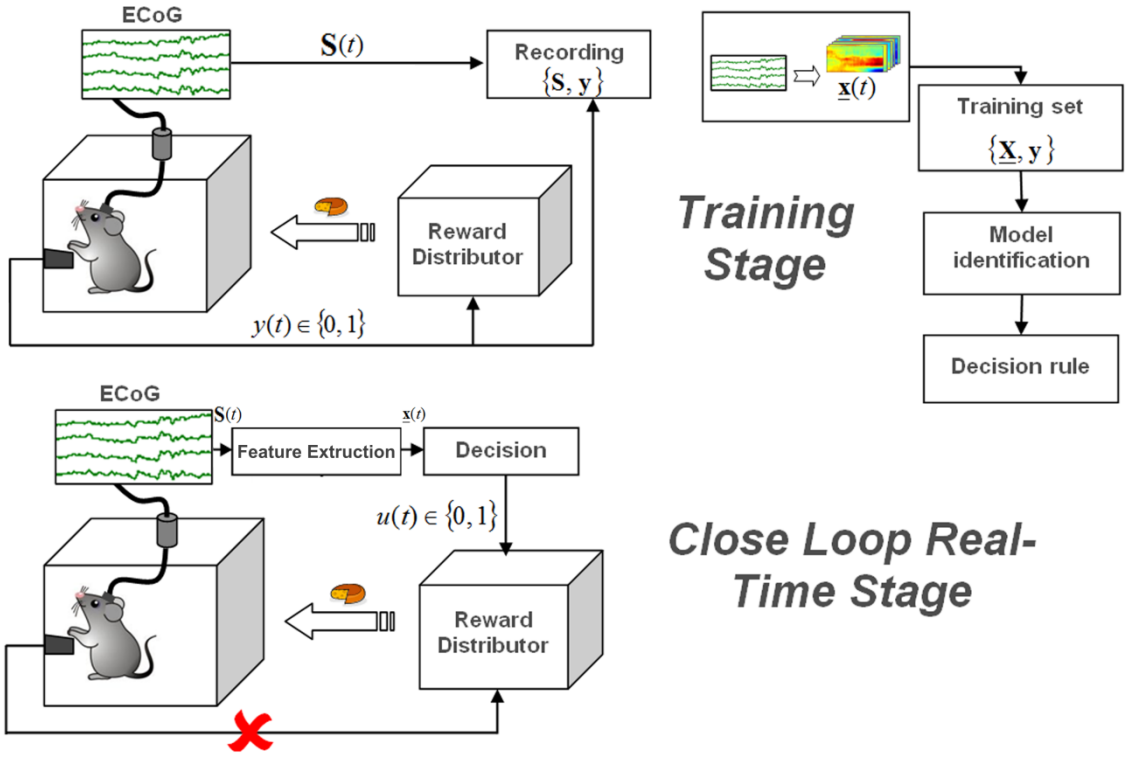

Figure 2.8 General scheme of the BCI experiments in rats. The training stage aims at the BCI system calibration resulting in a decision rule, which is applied immediately to the signal from the rat's brain during the Close Loop Real-Time experiments to activate the reward distributor. Here, $y(t)$ characterizes the state of the pedal at the moment t, **S** contains the signal from the brain recording during the experiment, $\underline{\mathbf{X}}$ is used for internal representation of the signal in the system, and $u(t)$ is a command signal generated by the system to control the food dispenser.

2.4 Experimental setup for non-human primates

Since experiments in monkeys were carried out to verify the approaches applied in rats, the schemes of the experiments are the same (Figure 2.9). During the first stage, namely, the Training Stage, the monkey can voluntary activate a fruit juice dispenser by pressing a pedal. After identification of the model, it is applied on the second stage to activate a reward distributor, whereas the pedal is disconnected.

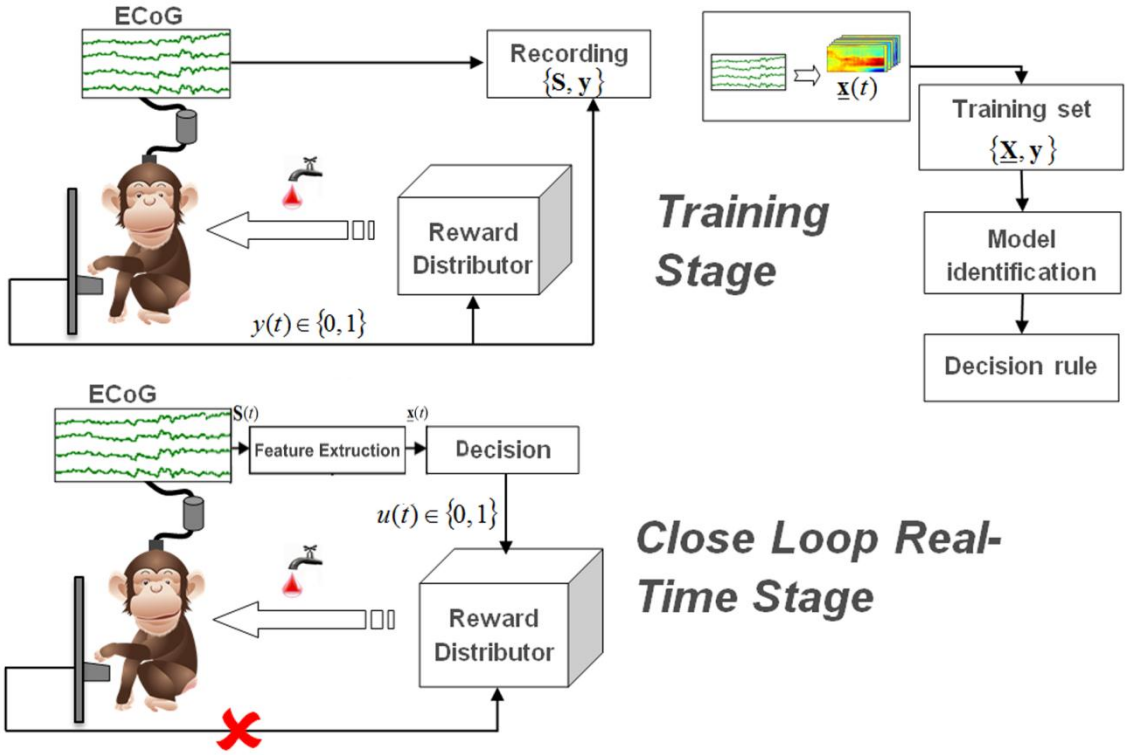

Figure 2.9 General scheme of the monkey's BCI experiments. The training stage aims at the BCI system calibration resulted in a decision rule, which is applied immediately to the signal from the monkey's brain during the Close Loop Real-Time experiments to activate the reward distributor. Here, $y(t)$ characterizes the position of the pedal at the moment t, **S** contains signal from the brain recording during the experiment, $\underline{\mathbf{X}}$ is used for internal representation of the signal in the system, and $u(t)$ is a command signal generated by the system to control the juice dispenser.

2.5 Conclusion

The BCI experiments were carried out in two animals, namely, the rat and the monkey. Neural activity of the animals' brains was recorded by means of the sets of ECoG electrodes which were preliminary implanted on the surfaces of their cortexes. To modulate the natural environment, the animal was either freely moving (the case of the rat), or minimally restrained during the experiments (the case of the monkey). During the experimental the animal had different types of activity to model the real-life conditions.

Chapter 3. BCI CONTROL SYSTEM

A general block-scheme of the BCI system consists in two main parts: the Control Block and the Object (Figure 3.1). The Control Block contains the Detector and the Controller BCI sub-blocks; the Object consists in the Effector (the pedal and the reward distributor in the Clinatec BCI experiment) and the Subject (a rat or a monkey). The Effector, characterized by its state (position of the pedal) $y(t) \in \{0,1\}$ at the moment t. It can affect the Subject according to the control signal $u(t)$ (the reward distributor gives a food pellet if $u(t) = 1$, otherwise $u(t) = 0$). The signals from the Subject $\mathbf{s}(t)$ (the recordings of the rat's or the monkey's brain activity) as well as information about the state of the Effector $y(t)$ are received by the Detector. During the first stage of the BCI experiment these data are used to estimate an unknown relation between the signals $\mathbf{s}(t)$ and $y(t)$. During the second stage, the Detector controls the Effector by means of the Controller BCI according to the estimated Effector's state $\hat{y}(t)$.

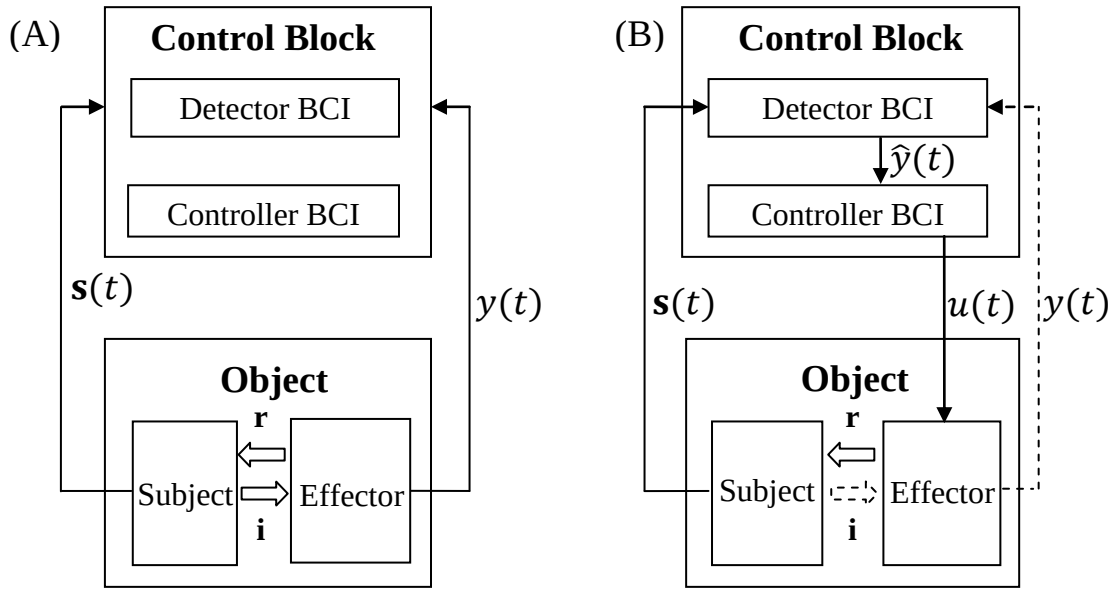

Figure 3.1 Block-scheme of the BCI Control Block. **(A)** corresponds to the training stage, **(B)** represents the second stage of the BCI system (the Close-Loop Real-Time Mode). $s(t)$ is the signal from the Subject at the moment t, $y(t)$ and $\hat{y}(t)$ characterize the real- and estimated state of the Effector, correspondently. $u(t)$ is the signal to control the Effector. The Subject interacts with the Effector: arrows **i** (influence) and **r** (response/feedback). In our experiments, the influence of the Subject on the Effector is a pressing event, whereas the response of the Effector to the Subject is reward distribution. On the second stage of the experiment **(B)**, the influence **i** is not taken into account (dotted arrow) and the response is determined by the Controller BCI, at the same time the Detector BCI no longer has an access to the state of the Effector $y(t)$ (dotted arrow).

3.1 Mathematical model of BCI system

Taking into account that the system is observed in the discrete moments of time $\{t_0, t_1, \dots\}$, let us denote $\mathbf{s}_i = \mathbf{s}(t_i)$, $y_i = y(t_i)$, and $u_i = u(t_i)$, for all $i = 0, 1, \dots$.

Mathematical model of the BCI system can be described as:

$$\begin{cases} y_i = f(\mathbf{s}_i, \dots, \mathbf{s}_0) + \varepsilon_i, \\ u_i = g(y_i, u_{i-1}, \dots, u_0), \end{cases} \tag{3.1}$$

here, $f(\cdot)$ is an unknown function, which defines relation between the recorded signal $\mathbf{s}(t)$ and the state of the Effector $y(t)$; $\varepsilon(t)$ is an additive noise; $g(\cdot)$ is a predefined

function, which on the base of the Effector state and previous control commands gives the new control command.

The unknown function $f(\cdot)$ is estimated during the training stage on the basis of the preliminary collected training set $S^{training} = \{\mathbf{s}_j^{training}, y_j^{training}\}_{j \in TrainingSet}$ (see Figure 3.1 (A)). The estimation $\hat{f}(\cdot)$ of the unknown function $f(\cdot)$ is applied at the second stage of the BCI experiment, namely, the Close-Loop Real-Time Mode (see Figure 3.1 (B)).

$$\begin{cases} \hat{y}_i = \hat{f}(\mathbf{s}_i, \dots, \mathbf{s}_0), \\ u_i = g(\hat{y}_i, u_{i-1}, \dots, u_0). \end{cases} \tag{3.2}$$

Finally, $\hat{y}_i$, which is the estimation of the Effector state y_i, is used in the Controller BCI (the predefined function $g(\cdot)$) to produce the control signal u_i (3.2).

3.2 Control Block

The Control Block is a part of the BCI system (Figure 1.1). The goal of the Control Block is to translate the Subject's brain activity $\mathbf{s}(t)$ into a control command for the Effector $\mathbf{u}(t)$ according to a predefined set of rules and previous states of the system. The Control Block consists in two sub-blocks: the Detector BCI and the Controller BCI (Figure 3.1).

3.2.1 Detector BCI

The Detector consists in the Signal Preprocessing, the Feature Extraction, the Model Calibration, and the Model Application (Figure 3.2). In addition, the Model Adaptation block could be included in the Detector.

Preprocessing Sub-Block

In general, the main purpose of the preprocessing is amplification of the internal signal, increasing of the signal-to-noise ratio, as well as artifacts processing (the Block either can remove artifacts from the signal or simply marks some epochs as artifact-contaminated). Several methods and approaches are frequently applied on this stage for ECoG/EEG data preprocessing:

- *Spectral Filtering*

Finite and Infinite Impulse Response Filters transform the source signal s into the filtered one s_{filt}:

$$a(1)s_{filt}(t) = b(1)s(t) + b(2)s(t-1) + \cdots + b(n_b)s(t-n_b-1)$$
$$-a(2)s_{filt}(t-1) - \cdots - a(n_a)s_{filt}(t-n_a-1),$$

for all t. Here, two sequences a and b with length n_a and n_b, respectively, can be calculated in several ways according to desired frequency band restrictions (Antoniou, 1993; Shenoi, 2006). This filter is called the Infinite Impulse Response Filter (IIR). In the special case of $n_a = 1$, the filter is called Finite Impulse Response Filter (FIR). IIR filters can produce steeper slopes between pass- and stop-bands, however, they can become unstable. FIR filters are always stable. Impulse Response Filters are widely used for the signal preprocessing in the BCIs (Dornhege et al., 2007; Zhang and Guan, 2010; Pistohl et al., 2008).

Another method used for the BCI's signal preprocessing is the *Fourier-Base Filter* (Dornhege et al., 2007). The method is based on the switching of the signal from the temporal to the spectral domain by calculating the short-time Fourier transform (STFT) (Oppenheim and Schafer, 1989). In the spectral domain the relevant frequencies are weighted and the Inverse Fourier transform (IFT) is applied.

- *Spatial Filtering*

  Together with the Spectral Filtering, the Spatial Filtering is one of the fundamental approaches for the BSI signals preprocessing. Bipolar Filtering, Common Average Reference, and Laplace Filtering are the most common methods for the Spatial Filters.

  To make the *Bipolar Filtering* all channels are recorded as voltage difference between electrode pairs (McFarland et al., 1997; Essl and Rappelsberger, 1998; Ramoser et al., 2000). Thus, the local voltage gradient is calculated, which emphasis a local activity while activity of the distant sources is suppressed.

  In the *Common Average Reference (CAR)* method the mean of all channels is subtracted from each individual channel (McFarland et al., 1997; Essl and Rappelsberger, 1998; Ramoser et al., 2000). CAR reduces influence of far sources,

but at the same time artifacts from one channel could be spread over all other channels.

Laplace Filtering consists in subtraction of the average of the surroundings signals from each individual channel (McFarland et al., 1997; Essl and Rappelsberger, 1998; Ramoser et al., 2000). The choice of the set of the surroundings electrodes defines the properties of the filtering.

In this study we have applied a band-stop filter (IIR notch filter) to cut off the 50 Hz noise (the frequency coming from the power supply) and a CAR filter (to reduce influence of the background noise). Applied methods provide satisfactory quality of the output data by means of significant elimination of irrelevant components of the signal. At the same time they are computationally efficient and fast enough to satisfy the severe restrictions imposed on the real-time BCI systems. Since quality of the ECoG signal is better than EEG one, artifact removal is absent on the preprocessing stage.

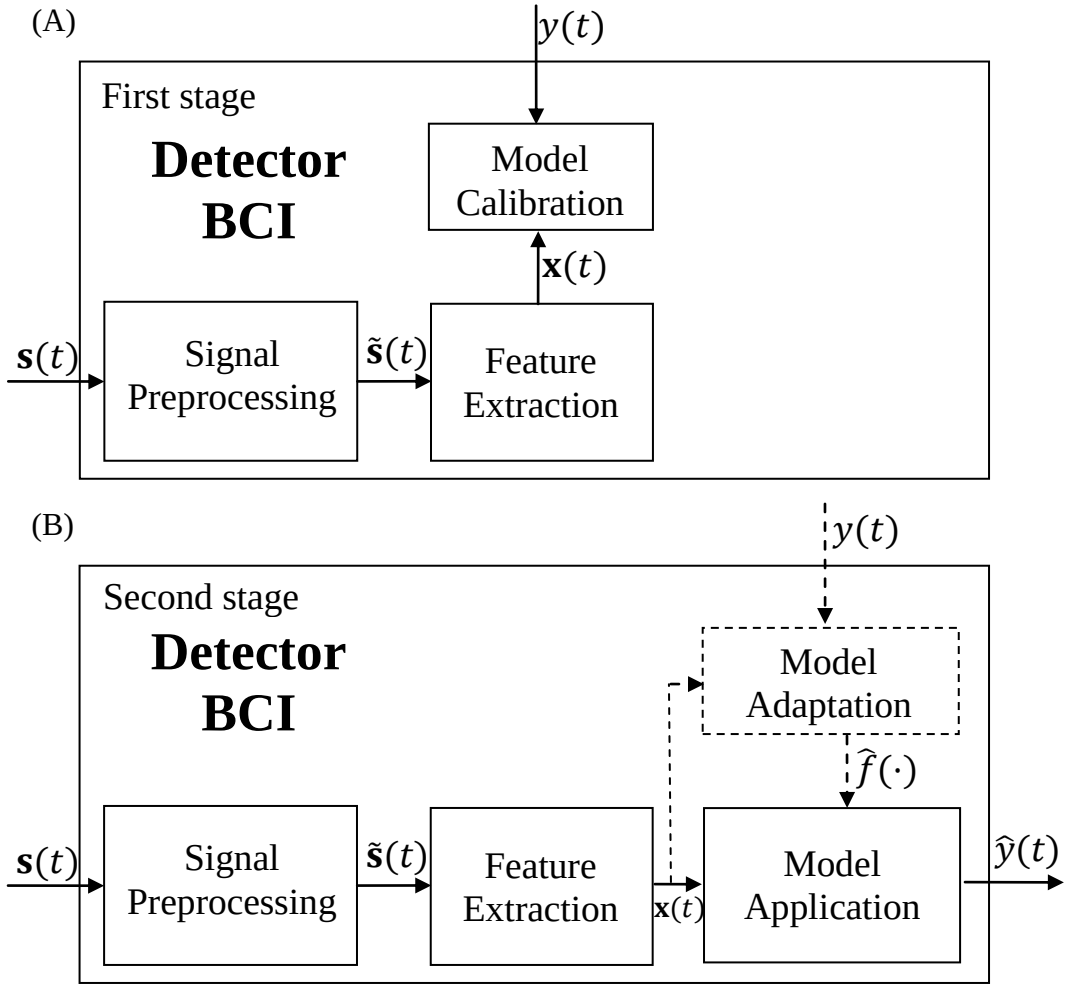

Figure 3.2 The general scheme the Detector Block. **(A)** During the first stage the signal from the Subject is used for the calibration of the model. **(B)** During the second stage the estimated model is used to evaluate the state of the Effector.

Feature Extraction Sub-Block

The purpose of the Feature Extraction sub-block is the mapping of the input signal to the feature space: $\mathbf{s}(t) \rightarrow \mathbf{x}(t)$. This feature space should be related to the underlying neurological phenomena correlated with the BCI task (e.g., user's intention of control). The most widespread methods for feature extraction in the ECoG based BCI studies are: time and/or frequency signal representation and parametric modeling (Bashashati et al., 2007b).

- *Time and/or frequency methods*

The most common method to estimate power-spectral density (PSD) features of the signal is the *Fast Fourier Transform (FFT)* (Sanei and Chambers, 2007), which has good computational efficiency and provides direct interpretation of the results. A drawback of the method is that it does not provide any time-domain information and, therefore, could not be efficiently applied for non-stationary signals. A windowed version of FFT (short-time Fourier transform, STFT) analyzes the time-varying spectral components of the data. The one-dimensional signal is mapped into a two-dimensional space of time and frequency. Unfortunately, frequency resolution is decreased, and estimated results do not reliably converge to the true PSD peaks (Tangermann, 2007). At the same time, the fixed time-window leads to limitation of the temporal resolution in the higher frequencies. Examples of the Fourier Transform applications in the BCI systems could be found in Kellis et al., 2010; Pfurtscheller et al., 2010b; Galán et al., 2008.

To overcome the temporal-resolution limitation of STFT, *Wavelet Transform (WT)* is often applied. The method maps the signal into a two-dimensional space of time and scales by means of the particular function, called the 'mother wavelet'. The choice the 'mother wavelet' is a crucial factor for WT. Like Fourier Transform, WT allows analyzing of both amplitude and phase components of the signal. For more information, see Section 5.2. Wavelet Transform is widely applied in BCI systems, for example see Chao et al., 2010, Fatourechi et al., 2008; Zhao et al., 2008; Hinterberger et al., 2003; Sherwood and Derakhshani, 2009; Cabrera et al., 2010.

The *Correlative Time-Frequency Representation (CTFR)* method provides information about the time-frequency interactions between the components of the input signal (Bashashati et al., 2007b) in addition to the spectral information. For instance, in the case of EEG, data samples are not analyzed independently, like in the case of FT or WT, but their relationship is also taken into account. However, the method is sensitive to noise (Garcia et al., 2003a; Garcia et al., 2003b).

- *Parametric modeling*

  The method assumes that the analyzed time series to be the output of a given linear model. The structure and order of the model should be chosen a priori (Weitkunat, 1991). For short signals parametric modeling results in better frequency resolution

in comparison with FT. However, estimation of the model's parameters is very sensitive to artifacts (Birch, 1988). A widely used example of the parametric model is the autoregressive (AR) model: $s(t) = -\sum_{k=1}^{p} \alpha(k)s(t-k)$, here p is the order of the model, and $\mathbf{A} = (\alpha(1), \dots, \alpha(p))$ are AR-coefficients. The vector of coefficients **A** is used further as the feature vector. Parametric models were applied for BCI, for instance, by Scherer et al., 2007; Kubánek et al., 2009.

In this study Continuous Wavelet Transform (CWT, see Section 5.2) is chosen, since this method allows efficiently revealing of underlying neurological phenomena correlated with BCI tasks. Unlike STFT, WT has no fixed time-window that leads to absence of limitation of the temporal resolution in the high frequencies. At the same time, contrary to discrete WT, continuous WT allows analyzing of the frequency range with better resolution.

Model Calibration Sub-Block

The goal of the Model Calibration sub-block is the estimation of the unknown function $f(\cdot)$ (see Equation (3.1)). For this purpose, during the first stage of the experiment (the calibration stage) the system records the signals $\mathbf{x}(t)$ and $y(t)$ to form the training set $S^{training} = \{\mathbf{x}_j^{training},\ y_j^{training}\}_{j \in TrainingSet}$. On the base of $S^{training}$ the function $\hat{f}(\cdot)$ is found in a class of functions F (e.g., linear, quadratic, polynomial functions, etc.) in such a way to minimize a functional $\Im$ (e.g., approximation error):

$$\Im\left(\{y_j^{training}, \hat{y}_j\}_{j \in TrainingSet}\right) \to \min_{\hat{f} \in F}, \tag{3.3}$$

where, $\hat{y}_j = \hat{f}\left(\mathbf{x}_{j-1}^{training}, \dots, \mathbf{x}_0^{training}\right)$.

Dimensionality reduction and/or feature selection (for instance, narrowing of the frequency range, estimation of the most informative channels, etc.) can be made preliminary (independently) or directly included in the optimization problem ((3.3).

In our case, as well as in the majority of other BCIs, $y(t)$ is a discrete variable. Mostly, different classifiers were applied to solve optimization problem (3.3), for instance, see a review of Lotte et al., (2007).

Linear Classifiers

Linear classifiers use linear functions to distinguish classes. The most used in the BCI tasks are Linear Discriminant Analysis (LDA, or Fisher's LDA) and Support Vector Machine (SVM).

- *Linear Discriminant Analysis*

  LDA assumes normal distribution of the data and uses hyperplanes to separate the data from the different classes (Fukunaga, 1990; Duda, et al., 2001). The method has low computational requirements, thus it could be effectively used for the online BCIs (Pfurtscheller, 1999; Bostanov, 2004; Garrett et al., 2003; Scherer et al., 2004). At the same time, on the data of complex non-linear structure the method has demonstrated poor results (Garcia et al., 2003b).

  To additionally penalize outliers and improve generalization capabilities of LDA regularization parameters could be introduced. A regularized Fisher's LDA (RFLDA) was used by Blankertz et al., (2002) and Müller et al., (2004).

- *Support Vector Machine*

  To identify classes SVM uses a discriminate hyperplane, selected in the way to maximize the distance between the nearest training points from the different classes (Cortes and Vapnik, 1995; Burges, 1998; Bennett and Campbell, 2000). SVM is known to have good generalization properties and to be steady to over-fitting effect (Jain et al., 2000). The method is widely applied in BCI (Rakotomamonjy et al., 2005; Garrett et al., 2003; Blankertz et al., 2002). At the same time, SVM allows natural generalization to the non-linear case by using the 'kernel trick' (Aizerman et al., 1964; Burges, 1998). Kernel SVM has also demonstrated good results in BCI applications (Kaper et al., 2004; Garrett et al., 2003).

Neural Networks

Neural Networks (NN) were used in BCIs for a long time (e.g., Hiraiwa et al., 1990). Among different NN the most widespread is the Multilayer Perceptron (MLP).

- *Multilayer Perceptron*

  MLP is an assembly of several layers of artificial neurons: an input layer, hidden layers, and an output layer. When composed of enough neurons MLP can

approximate any continuous function (Bishop, 1996). The negative side of this property is its sensitivity to overtraining, especially for such noisy and instable data like brain's neuronal activity (Balakrishnan and Puthusserypady, 2005). A MLP without any hidden layer is called a perceptron and it could be shown that it is equivalent to LDA.

MLPs were applied in variety of BCI tasks: binary (Palaniappan, 2005), multiclass (Anderson and Sijercic, 1996), synchronous (Haselsteiner and Pfurtscheller, 2000), and asynchronous (Chiappa and Bengio, 2004). A perceptron without hidden layers were used by Wang et al., (2004) and Congedo et al., (2006).

- *Other Neural Networks Architectures*

  Other Neural Networks used in BCI systems are: *Gaussian classifier NN* (Millán et al., 2000; Millán et al., 2004; Solhjoo and Moradi, 2004; Cincotti et al., 2003), *Learning Vector Quantization NN* (Kohonen, 1990; Pfurtscheller et al., 1993), *fuzzy ARTMAP NN* (Carpenter et al., 1992; Palaniappan et al., 2002), *dynamic NN* (Haselsteiner and Pfurtscheller, 2000; Barreto et al., 1996), *RBF NN* (Duda et al., 2001; Hoya et al., 2003), *Bayesian Logistic Regression NN* (BLRNN) (Penny et al., 2000), Adaptive Logic Network (ALN) (Kostov and Polak, 2000), *Probability Estimating Guarded Neural Classifier* (PeGNC) (Felzer and Freisieben, 2003).

Non-Linear Classifiers

Non-linear classifiers uses non-linear decision boundaries therefore they can more efficiently reject uncertain samples. At the same time, they are more sensitive to overtraining, more resource consumption, and therefore they are less often applied in BCI systems.

- *Bayes Quadratic*

  In this approach the Bayes rule is used to compute *a posteriori* probability that a future vector belongs to a given class (Duda et al., 2001; Fukunaga, 1990). The class of the feature vector is estimated by the Maximum *a posteriori* probability (MAP) rule. It is assumed that the data in the different classes have different normal distribution, which leads to quadratic decision boundary. The method was applied

by Lemm et al., (2004), Solhjoo and Moradi, (2004), Keirn and Aunon, (1990), Barreto et al., (2004).

- *Hidden Markov Model*

  Hidden Markov Model (HMM) is a statistical model in which the system is assumed to be a Markov process with unobserved (hidden) states (Rabiner, 1989). HMM are efficiently used for classification of the time series. HMM were realized in BCI systems by Obermeier et al., (2001), Cincotti et al., (2003), Obermaier et al., (2000), Solhjoo et al., (2005), Chiappa and Bengio, (2004).

Nearest neighbor Classifiers

Classifiers of this type assign a feature vector to a class according to its nearest neighbors. These classifiers are non-linear.

- *k-Nearest Neighbors*

  A feature vector is assigned to the dominant class among the k nearest neighbors in the training set (Duda et al., 2001). The algorithm was applied in a set of BCIs (Blankertz et al., 2002; Müller et al., 2004; Schlögl et al., 2005), however it is too sensitive to the curse-of-dimensionality (Friedman, 1997). At the same time, in the case of BCI with low dimensional feature vectors, kNN was efficient (Borisoff et al., 2004).

- *Mahalanobis Distance*

  The classifier assumes a Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$ for each prototype of the class c. A feature vector $\mathbf{x}$ is assigned to the class that corresponds to the nearest prototype according to the Mahalanobis distance $d_c(\mathbf{x})$ (Cincotti et al., 2003): $d_c(\mathbf{x}) = \sqrt{(\mathbf{x}-\boldsymbol{\mu}_c)\boldsymbol{\Sigma}_c^{-1}(\mathbf{x}-\boldsymbol{\mu}_c)^T}$. The method gives simple and robust classifier which was used for the BCI systems by Schlögl et al., (2005) and Cincotti et al., (2003).

Combinations of Classifiers

A recent trend in the BCI systems consists in combination of different classifiers. There are several strategies of combination. Here, we are referring to the most popular.

- *Boosting*

  Several classifiers are applied in cascade, each classifier is focusing on the errors committed by the previous ones (Duda et al., 2001). Thus, from the several weak classifiers, a powerful one could be constructed. At the same time, it will not be sensitive to overtraining. From the other hand, the method is sensitive to mislabels (Jain et al., 2000). Boosting was applied in BCI by Boostani and Moradi, (2004); Hoffmann et al., (2005).

- *Voting*

  Several classifiers are applied independently and final decision will be that of majority (Jain et al., 2000). Voting is very popular way of classifiers combination (Pfurtscheller et al., 1993; Rakotomamonjy et al., 2005; Qin et al., 2005).

- *Stacking*

  Two levels of classifier are used: level-0 and level-1.The outputs of a set of independent level-0 classifiers are used as an input for a level-1 (meta) classifier, which makes a final decision (Wolpert, 1992). The method was efficiently applied in the BCI research by Lee and Choi, 2003.

Regression

Although classifiers are used in the majority of the BCIs, regression methods are also applied to solve optimization problem (3.3) (McFarland et al., 2006; Kubánek et al., 2009; Chao et al., 2010). Generally regression approaches are used for continuous values of $\mathbf{y}(t)$. For instance, $\mathbf{y}(t)$ can describe motion of object in the space (see Kubánek et al., 2009; Chao et al., 2010). However, the regression methods can be applied also for the discrete case (McFarland et al., 2006). Regression and classification algorithms for the discrete BCI control demonstrate comparative results (McFarland et al., 2006). In our study we utilize a linear regression approach with simultaneous projection to a low dimensional feature space and further binarization. The regression was chosen because it allows natural generalization from the case of the binary response variable $y(t) \in \{0,1\}$ to the discrete ($\mathbf{y}(t) \in \{0,1\}^M$, $M \geq 1$) or continuous vector $\mathbf{y}(t) \in \mathbb{R}^M$, $M \geq 1$, which correspond to the discrete multiclass or continuous control.

They are the perspectives of our study. In addition, time-lag information can be easily introduced in both dependent and independent variables.

More precisely the proposed method will be discussed in the next chapter.

Model Application Sub-Block

Founded during the first stage of the experiment estimation $\hat{f}(\cdot)$ is applied to the signal in the Model Application sub-block during the second stage of the BCI experiment. As a result, the estimation of the system state $\hat{y}(t)$ is generated and sent as an output of the Detector BCI to the Controller BCI.

Model Adaptation Sub-Block

The Model Adaptation sub-block is intended for adaptation of the BCI system to the different types of changes in the signal over time due to the brain plasticity or other external reasons. During the adaptation stage the sub-block has a temporal access to $y(t)$, which describes the real state of the Effector (in Figure 3.2 (B) shown as a dashed arrow). The Model Adaptation sub-block activates occasionally and adjusts the function $\hat{f}(\cdot)$ to minimize a misalignment between the Effector state estimation $\hat{y}(t)$ and its real state $y(t)$.

3.2.2 Controller BCI

The Controller BCI translates the features into the control signals by means of the preliminary established function $g(\cdot)$ (see Equation (3.1)). In the case of the binary BCI, the generated control signal $u(t)$ is logical: 0 denotes the NC states (the 'non-event' situations when no food is given) and 1 indicates the IC states (the 'event' detection situations when food is given). The sub-block includes a post-processing of the output control signal to reduce the number of activation of the system (for instance, it could block the system for some time after activation).

Chapter 4. MULTI-WAY ANALYSIS FOR BCI SYSTEMS

4.1 Introduction

A common approach for the brain signal processing, intended for event detection/prediction, consists in the extraction of the event-related features of the neuronal activity. Information from spatial (Rakotomamonjy et al., 2005), frequency (Schlögl et al., 2005), and temporal (Vidaurre et al., 2009) domains could be analyzed by means of the Principal Component Analysis (PCA) (Kayser et al., 2003), the Independent Component Analysis (ICA) (Makeig et al., 1999), the Linear Discriminant Analysis (LDA) (Scherer et al., 2004), the Common Spatial Patterns (CSP) (Zhao et al., 2008), the Partial Least Squares (PLS) (Chao et al., 2009), etc. Let us note that the standard methods are designed mainly for vector input variables which present either one domain of analysis or several domains unfolded in one line. Using only one domain usually does not provide satisfactory results. Combination of several domains is thus necessary. In most cases, two or three ways of analysis are applied sequentially. For example, see Galán et al., (2008), where first stable frequency components are determined and second the best electrodes are chosen. A multi-way analysis allows simultaneous treatment of several domains, by means of a tensor-based data representation. In recent years, it was applied in several BCI studies and demonstrated promising results (e.g., tensor factorization with PARAFAC (Nazarpour et al., 2006), Tucker (Zhao et al., 2009), Non-negative Tensor Factorization (Mørup et al., 2008), Multi-way Partial Least Squares (NPLS) (Bro, 1996), General Tensor Discriminant Analysis (Li et al., 2009), Regularized Tensor Discriminant Analysis (Li and Zhang, 2010)). Therefore, it is chosen for the ECoG data representation in the current study.

The signals from the different electrodes are mapped by a continuous wavelet transform to the temporal-frequency-spatial space. As a result, every time epoch is

represented by a cube (a third-order tensor). All cubes are stored in a tensor of observations (a fourth-order tensor).

The Multi-way Partial Least Squares (N-way Partial Least Squares, NPLS) (Bro, 1996) presents an effective approach for tensor data analysis. Namely, NPLS is a statistical method for a linear regression identification in the case of tensor input or/and output variables. It is particularly suited for high dimensional data. While NPLS works without any prior knowledge, it can be efficiently applied for automatic identification of the model aimed at prediction of the BCI events from the measurement of the brain neuronal activity. In comparison to the ordinary Partial Least Squares or other vector oriented algorithms that can be applied to the tensor data after unfolding, NPLS preserves the multimodality of the data in a robust way (Bro, 1996). In addition, it allows identifying how much each feature contributes to the decision rule (for instance, which electrode is the most informative). NPLS is based on tensor factorization and data projection to a low dimensional feature space. In comparison to other tensor-based methods that were previously applied to BCI studies recently, NPLS involves supervised tensor decomposition which significantly increases the efficiency of modeling. At the same time, whereas others methods are restricted to discriminative task, NPLS can be applied for both classification and regression. Besides that, NPLS was successfully used for epilepsy seizure recognition (Acar et al., 2007). That is why it was chosen as the basic approach in our work. However, the NPLS algorithm has a set of drawbacks connected with possible excessive complexity of the identified model, method's computational efficiency and resources consumption. To overcome these drawbacks, penalized, iterative and recursive versions of the algorithm are proposed, which allow informative subsets selection, huge datasets processing, and online adaptation, correspondently. Detailed description of the NPLS approach as well as of its modifications are given in this chapter.

4.2 Tensor notations and preliminaries

In this section, some background information about tensors is given. For more details see, for example, Kolda and Bader (2007) or Comon (2002).

Tensors (multi-way arrays) are higher-order generalization of vectors and matrices (this notion of tensors must be distinguished from tensors in physics (e.g.,

stress tensors), which are generally referred to as tensor fields in mathematics). Let us denote tensors by boldface underlined capital or lowercase letters ($\underline{\mathbf{X}}$, $\underline{\mathbf{x}}$), matrices are represented by boldface capital letters ($\mathbf{X}$), vectors are denoted by boldface lowercase letters ($\mathbf{x}$), and for scalars, we will use lowercase letters (x). An element of a tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ is referred to as $x_{i_1, i_2, \ldots, i_N}$. Here, N is the order of the tensor, i.e., its number of dimensions, also known as ways or modes. Vectors and matrices are tensors of order one and two, respectively. The number of the variables I_i in the i-th mode shows dimensionality of this mode (indices typically range from 1 to their capital version, e.g., $i_5 = 1, \ldots, I_5$). An example of a third-order tensor is given in Figure 4.1.

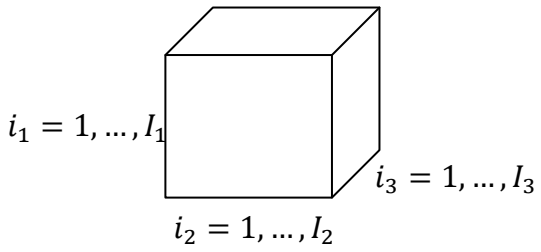

Figure 4.1 A third-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$.

4.2.1 Rank of tensor. Rank-one tensors

A tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ always admits a decomposition into a sum of outer products as (Comon and Berge, 2006):

$$\underline{\mathbf{X}} = \sum_{r=1}^{R} \mathbf{a}_r^1 \circ \mathbf{a}_r^2 \circ \ldots \circ \mathbf{a}_r^N, \tag{4.1}$$

where for each r the vector $\mathbf{a}_r^j \in \mathbb{R}^{I_j}$, and "$\circ$" denotes the outer product. For more information see Kolda and Bader (2007).

Each tensor's element of Equation (4.1) could be written as:

$$x_{i_1 i_2 \ldots i_N} = \sum_{r=1}^{R} a_{r,i_1}^1 a_{r,i_2}^2 \ldots a_{r,i_N}^N, \qquad \forall\, i_j = 1, \ldots, I_j, \qquad j = 1, \ldots, N, \tag{4.2}$$

where, a_{r,i_j}^j is the i_j-element of the vector $\mathbf{a}_r^j \in \mathbb{R}^{I_j}$.

The rank of the given tensor $\underline{\mathbf{X}}$ is the minimal integer number R such that this decomposition (4.1) is exactly satisfied (Comon and Berge, 2006).

The tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ is *rank-one* if $\exists \{\mathbf{a}^1 \in \mathbb{R}^{I_1}, \ldots, \mathbf{a}^N \in \mathbb{R}^{I_N}\}$, such that it is possible to represent the tensor as:

$$\underline{\mathbf{X}} = \mathbf{a}^1 \circ \mathbf{a}^2 \circ \ldots \circ \mathbf{a}^N \tag{4.3}$$

In the current manuscript graphically the rank-one tensors are represented as shown in Figure 4.2.

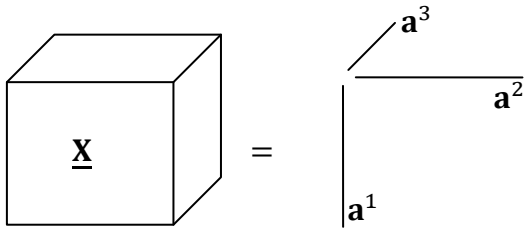

Figure 4.2 The third-order rank-one tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$: $\underline{\mathbf{X}} = \mathbf{a^1} \circ \mathbf{a^2} \circ \mathbf{a^3}$.

For example, the tensor $\underline{\mathbf{X}}$: $\underline{\mathbf{X}}_{:,:,1} = \begin{pmatrix} -4 & -6 \\ 4 & 6 \end{pmatrix}$, $\underline{\mathbf{X}}_{:,:,2} = \begin{pmatrix} 2 & 3 \\ -2 & -3 \end{pmatrix}$ is the rank-one tensor: $\underline{\mathbf{X}} = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \circ \begin{pmatrix} 2 \\ 3 \end{pmatrix} \circ \begin{pmatrix} -2 \\ 1 \end{pmatrix}$.

4.2.2 Tensor operations

- *Unfolding (tensor → matrix)*

Tensor unfolding, also known as matricization or flattening, is the mapping of a tensor into a matrix. The mode-n unfolding $\underline{\mathbf{X}}_{(n)}$ of a tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ arranges the mode-n fibers (see Figure 4.4) to be the columns of the matrix. Namely, a tensor element $x_{i_1, i_2, \ldots, i_N}$ is mapped to a matrix element $x_{i_n, j}$: $j = 1 + \sum_{k=1, k \neq n}^{N} (i_k - 1) J_k$, where $J_k = \prod_{m=1, m \neq n}^{k-1} I_m$ ($J_k = 1$ if $k = 1$, $J_k = 1$ if $k = 2$ and $n = 1$). The unfolding operation is illustrated for a third-order tensor in Figure 4.3.

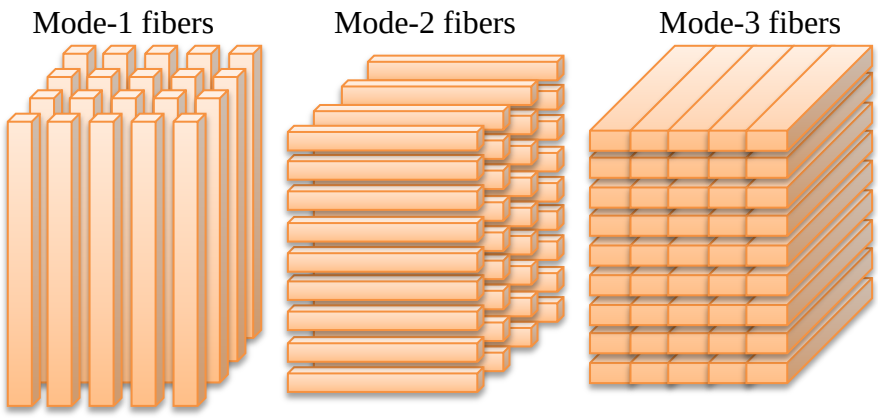

Figure 4.4 Fibers for a third-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$.

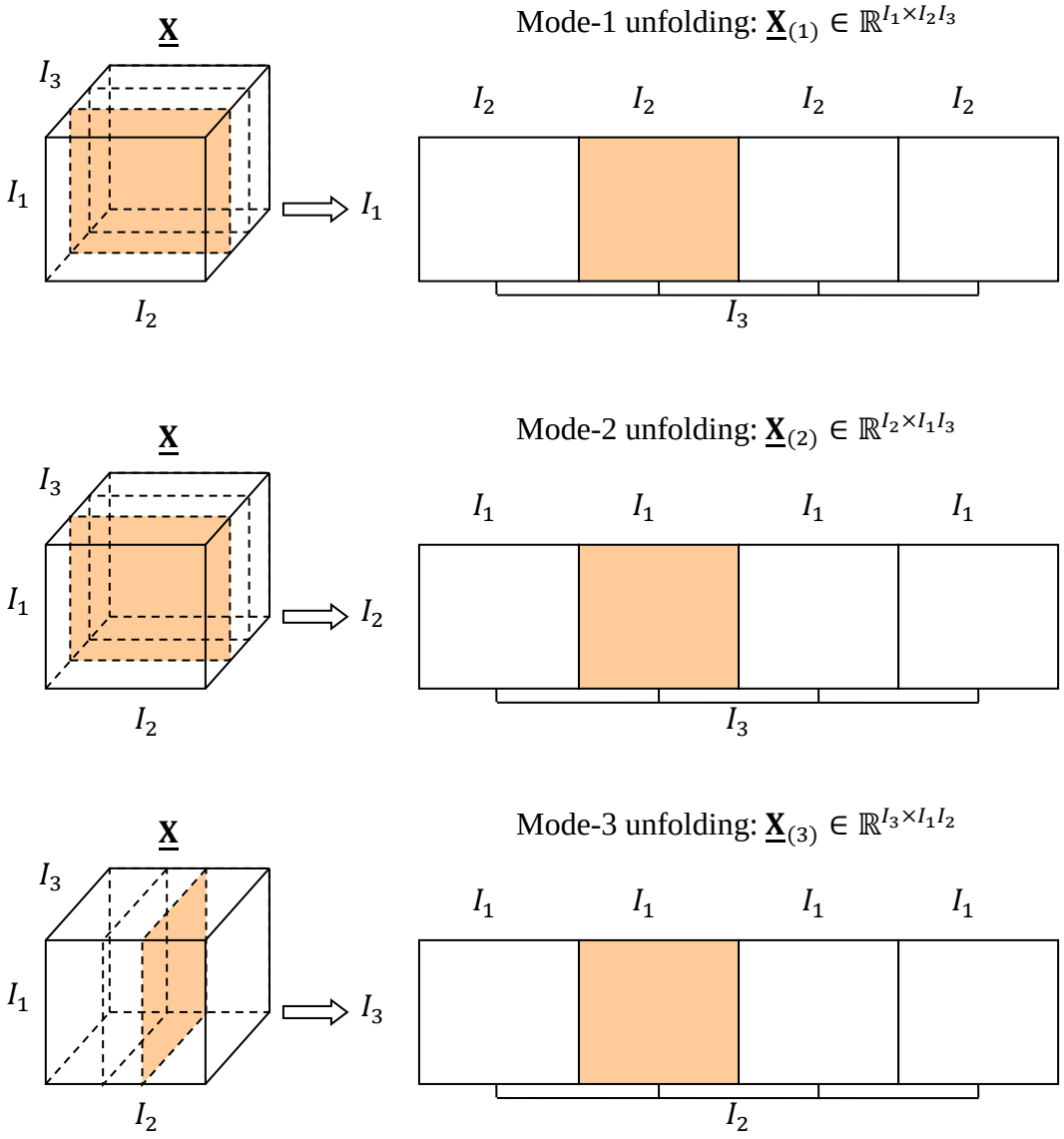

Figure 4.3 Different ways to unfold a third-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$.

- *Vectorization (matrix → vector)*

Vectorization of a matrix $\mathbf{X}$, denoted as vec($\mathbf{X}$), is a column-wise unfolding of $\mathbf{X}$ into a column vector. For example, if $\mathbf{X} = [\mathbf{x}_1 | \ldots | \mathbf{x}_n]$ then

$$\text{vec}(\mathbf{X}) = \begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_n \end{bmatrix}.$$

- *Norm*

The norm of a tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ is generalization of the Frobenius norm:

$$\left\| \underline{\mathbf{X}} \right\|_F = \sqrt{\sum_{i_1=1}^{I_1} \ldots \sum_{i_N=1}^{I_N} \left| x_{i_1 \ldots i_N} \right|^2} \tag{4.4}$$

- *n-mode (vector) product*

The n-mode (vector) product of a tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ and a vector $\mathbf{v} \in \mathbb{R}^{I_n}$ is denoted as $\underline{\mathbf{X}} \times_n \mathbf{v}$. The result tensor has $(N-1)$ order and its size is $I_1 \times \ldots \times I_{n-1} \times I_{n+1} \times \ldots \times I_N$. Elementwise,

$$\left(\underline{\mathbf{X}} \times_n \mathbf{v} \right)_{i_1 \ldots i_{n-1} i_{n+1} \ldots i_N} = \sum_{i_n=1}^{I_n} x_{i_1 \ldots i_N} v_{i_n} \tag{4.5}$$

4.3 Multi-way representation of ECoG data

To form the BCI feature tensor, the ECoG signal (see Figure 2.3 and Figure 2.7) is mapped to the temporal-frequency-spatial space by Continuous Wavelet Transform (CWT) (Figure 4.5). The observation tensor $\underline{\mathbf{X}}$ is formed from a set of epochs, such as for each epoch j (determined by its final moment t_j), electrode c, frequency f and time shift τ, elements $x_{j,\tau,f,c}$ of the tensor were calculated as absolute value of CWT coefficients of the ECoG signal.

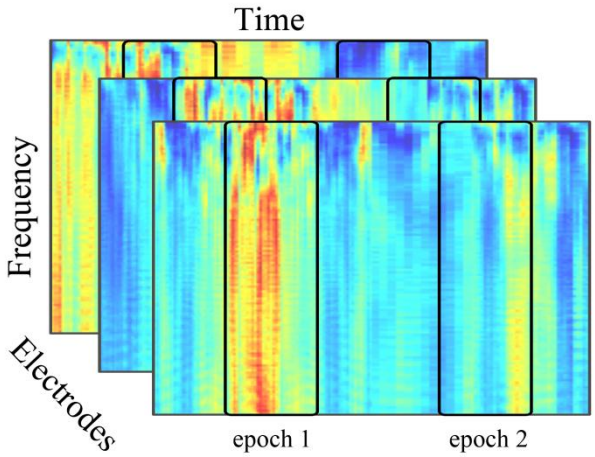

Figure 4.5 The multi-channel ECoG recording is mapped to the temporal-frequency-spatial feature space.

The size of the tensor $\underline{\mathbf{X}}$ (multiplication of tensor's dimensions) can be very large. For instance, 14 electrodes are used in our experiments in rat, 32 electrodes are implanted on the cortex of monkey, and up to 128 are foreseen for the future human implant. At the same time, frequency and time modalities should provide appropriate data resolution (in our analysis, dimensions are equal to 146 and 51, correspondently). Therefore, the size of the observation point $\underline{\mathbf{x}}$ is equal to $14 \cdot 146 \cdot 51 \approx 10^5$ (rat experiments). During training we use up to 1400 epochs. Thus, the size of the observation tensor $\underline{\mathbf{X}}$ is $14 \cdot 146 \cdot 51 \cdot 1400 \approx 10^8$.

4.4 Statement of problem of BCI system calibration from experimental data

The goal of the Calibration stage in BCI experiments consists in the determination of dependency between the brain neural activity signal (an observed multivariate variable) and the state of the system (a response binary variable). Let us denote the explanatory variable as $\underline{\mathbf{x}}(t)$ and the response variable as $\mathbf{y}(t)$. A model of $\mathbf{y}$ as a function of $\underline{\mathbf{x}}$ is estimated during the calibration stage, by solving the optimization problem (3.3). To this purpose training data $\{\underline{\mathbf{X}}, \mathbf{Y}\}$ are formed from n observed data samples. In the case of regression, the minimization functional $\Im$ is often taken as sum of squares of residual norms:

$$\sum_{j \in TrainingSet} \left\| \mathbf{y}_j^{training} - \hat{\mathbf{y}}_j \right\|^2 \to \min_{\hat{f} \in F},$$
$$\hat{\mathbf{y}}_j = \hat{f}\left(\underline{\mathbf{x}}_j^{training}, \dots, \underline{\mathbf{x}}_0^{training}\right). \quad (4.6)$$

In the present study, specificity of optimization task (4.6) consists in tensor representation of the data and high dimensionality (more variables than observations). Moreover, the variables are highly correlated. The methods, applied for the model identification, should take into account these peculiarities. In particular, to reduce the dimensionality a set of projections, saving tensor data structure, should be applied. The regression is determined either simultaneously with projecting or after it.

4.5 Two-way modeling

Before the description of regression approaches for tensor data, we consider the matrix methods in the case of large dimension and highly correlated variables. Let $\mathbf{x} \in \mathbb{R}^N$ and $\mathbf{y} \in \mathbb{R}^M$ be vectors of observations of explanatory and response variables, respectively. Observation of n data samples gives matrices $\mathbf{X} \in \mathbb{R}^{n \times N}$ and $\mathbf{Y} \in \mathbb{R}^{n \times M}$. Two projection-based methods, namely, Principal Component Analysis (PCA)/ Principal Component Regression (PCR) and Partial Least Squares Regression (PLS), are often used for dimension reduction and regression identification.

4.5.1 Principal Component Analysis (PCA) and Principal Component Regression (PCR)

Principal Component Analysis is a standard approach to transform a number of possibly *correlated* observed variables into a smaller number of *uncorrelated* variables called *principal components*. With this transformation, the first principal component accounts for as much of variability in the data as possible, and each following component accounts for as much of the remaining variability as possible. Algebraic decomposition of the matrix of the observed variables $\mathbf{X} \in R^{n \times N}$ could be expressed as

$$\mathbf{X} = \sum_{f=1}^{F} \mathbf{t}_f \mathbf{p}_f^T + \mathbf{E} = \mathbf{T}\mathbf{P}^T + \mathbf{E},$$
$$\|\mathbf{E}\|_F \to \min \quad (4.7)$$

where, $\mathbf{T} = [\mathbf{t}_1 | \dots | \mathbf{t}_F]$ is a matrix of the $F \leq n$ extracted score vectors (vectors of latent variables), $\mathbf{P} = [\mathbf{p}_1 | \dots | \mathbf{p}_F]$ represents a matrix of the extracted loadings, and $\mathbf{E}$ is a

matrix of residuals. In addition, prior to application of the PCA decomposition, the variables in **X** must be column mean-centered.

Graphical representation of the PCA approach is shown in Figure 4.6.

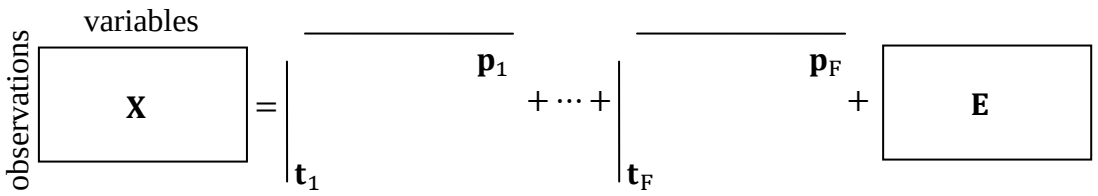

Figure 4.6 Graphical representation of the PCA decomposition of a matrix **X**.

Each consecutive principal component is chosen to maximize the decrease of variation of the matrix **X**. It can be shown that in PCA the loading vectors $\mathbf{p}_f$, $f = 1, \ldots, F$ are the sorted eigenvectors of the matrix $\mathbf{X}^T\mathbf{X}$, whereas the scores $\mathbf{t}_f$, $f = 1, \ldots, F$ are the sorted eigenvectors of the matrix $\mathbf{X}\mathbf{X}^T$ (Geladi and Kowalski, 1986).

Thus, PCA is theoretically the optimal linear scheme, in terms of the least mean square error, for compressing a set of high dimensional vectors into a set of lower dimensional vectors. The solution defined by PCA is not unique, there is a rotational freedom. Therefore, it should be made some type of post-processing to select the most appropriate decomposition for the task.

After the principal components for the matrix **X** have been found, a regression **Y** on **T** can be identified. For example, it could be a linear regression (PCR) $\widehat{\mathbf{Y}} = \mathbf{T}\widehat{\mathbf{B}}^T$. In this case, the least square solution gives $\widehat{\mathbf{B}}^T = (\mathbf{T}^T\mathbf{T})^{-1}\mathbf{T}^T\mathbf{Y}$.

4.5.2 Partial Least Squares Regression (PLS)

Partial Least Squares (PLS) regression is a statistical method for vector-based data analyses, particularly appropriate to the case of high dimension (Geladi and Kowalski, 1986). As opposed to other widely used projection-based methods, like PCA, PLS uses for factorization not only the matrix of the independent variables **X** but also the matrix of the dependent variables **Y**, this increases the efficiency in selection of the most relevant to **Y** features. A PLS model tries to find such principal components to

explain the maximum variance of **X** and **Y** simultaneously. Algebraically it could be shown as:

$$\begin{gathered}\mathbf{X}=\sum_{f=1}^{F}\mathbf{t}_f\mathbf{p}_f^T+\mathbf{E}=\mathbf{T}\mathbf{P}^T+\mathbf{E},\\ \mathbf{Y}=\sum_{f=1}^{F}\mathbf{u}_f\mathbf{q}_f^T+\mathbf{F}=\mathbf{U}\mathbf{Q}^T+\mathbf{F},\\ \operatorname{cov}(\mathbf{t}_f,\mathbf{u}_f)\rightarrow\max,\ f=1,\ldots,F.\end{gathered} \tag{4.8}$$

where, $\mathbf{T}=[\mathbf{t}_1|\ldots|\mathbf{t}_F]$, $\mathbf{U}=[\mathbf{u}_1|\ldots|\mathbf{u}_F]$ are matrices of the $F\leq n$ extracted score vectors, $\mathbf{P}=[\mathbf{p}_1|\ldots|\mathbf{p}_F]$ and $\mathbf{Q}=[\mathbf{q}_1|\ldots|\mathbf{q}_F]$ represent matrices of the extracted loadings, and **E**, **F** are matrices of residuals. In addition, like in the case of PCA, prior to application of the PLS decomposition, the variables in **X** and **Y** must be column mean-centered.

Graphical representation of the PLS approach is shown in Figure 4.7.

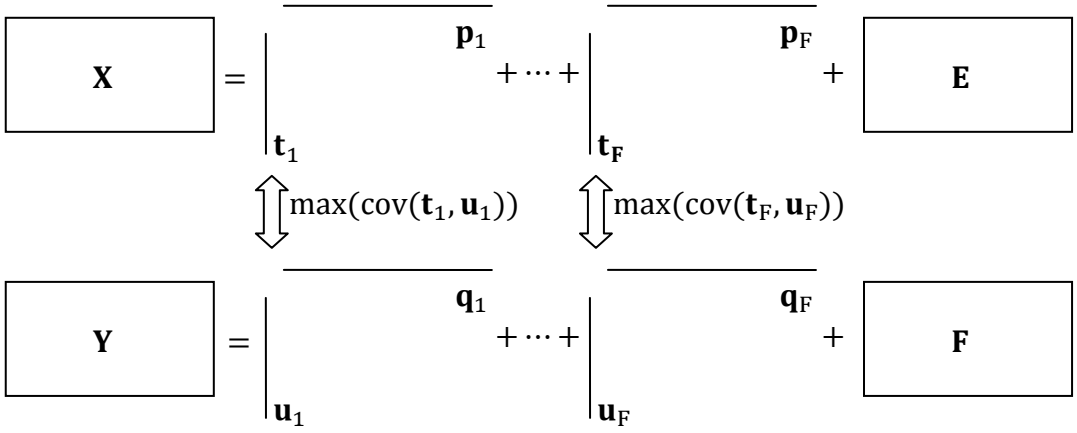

Figure 4.7 Graphical representation of the PLS decomposition of the matrices **X** and **Y**.

Finally, PLS properly handles situations when the matrix of observations **X** contains more variables N than observations n and the variables are highly correlated (Geladi and Kowalski, 1986).

4.5.3 Recursive Partial Least Squares Regression (RPLS)

Several Recursive PLS algorithms were invented to take into account time-dependent changes of data as well as to be able to handle large data sets. Qin (1998) and Dayal and MacGregor (1997) introduced the most known approaches. The method introduced by Dayal and MacGregor has better performance and does not suffer from problems concerned with short data windows (for more information, see Dayal and MacGregor, 1997). However, Dayal's algorithm stores in the active memory the covariance matrix $\mathbf{X}^T\mathbf{X}$, which dimension is equal to square of **x**. This requirement is excessively difficult for high dimensionality tasks. In our study, data dimensionality is extremely large ($\sim 10^5$). Therefore, we focus ourselves on Qin's algorithms (Qin, 1998).

According to Qin's algorithm, the matrices **X** and **Y** are decomposed by the batch-wise PLS algorithm with orthonormal latent variables. In general, the PLS model is constructed as:

$$\mathbf{X} = \mathbf{T}\mathbf{P}^T + \mathbf{E}, \mathbf{Y} = \mathbf{T}\mathbf{B}\mathbf{Q}^T + \mathbf{F}, \tag{4.9}$$

where $\mathbf{B} = diag\{b_1, \dots, b_F\}$. The matrix of latent variables **T** is orthonormal (see Qin, 1998):

$$\mathbf{T}^T\mathbf{T} = \mathbf{I}_F, \tag{4.10}$$

where $\mathbf{I}_F$ is an F by F identity matrix. In addition, $\mathbf{T}$ is orthogonal to both residuals matrices:

$$\mathbf{T}^T\mathbf{E} = \mathbf{0}, \tag{4.11}$$

$$\mathbf{T}^T\mathbf{F} = \mathbf{0}, \tag{4.12}$$

In his work, Qin has shown that if the number of factors F is large enough to provide the condition $\mathbf{E}^T\mathbf{F} = 0$, then $\mathbf{X}^T\mathbf{X} = \mathbf{P}\mathbf{P}^T$ and $\mathbf{X}^T\mathbf{Y} = \mathbf{P}\mathbf{B}\mathbf{Q}^T$. This yields that for the new data pair $\{\mathbf{X}_{new}, \mathbf{Y}_{new}\}$, the regressions on the next data sets will be equivalent:

$$\left\{\begin{bmatrix}\mathbf{X}\\ \mathbf{X}_{new}\end{bmatrix}, \begin{bmatrix}\mathbf{Y}\\ \mathbf{Y}_{new}\end{bmatrix}\right\} \Leftrightarrow \left\{\begin{bmatrix}\mathbf{P}^T\\ \mathbf{X}_{new}\end{bmatrix}, \begin{bmatrix}\mathbf{B}\mathbf{Q}^T\\ \mathbf{Y}_{new}\end{bmatrix}\right\}. \tag{4.13}$$

Thus, the old data **X** and **Y** are captured by the loading matrices **P** and **Q**, as well as by the coefficient matrix **B**, whereas the new data are concatenated to their combination. As the result, the algorithm always keeps the size of the stored in the active memory datasets. In addition, this approach represents an effective tool for the adaptive learning, by introducing weights for the old and new data (see Qin, 1998).

4.6 Generic multi-way approaches

PCA and PLS methods were originally designed for the vector input variables. In practice for a wide variety of tasks several domains must be analyzed simultaneously to obtain proper results (e.g., time-series analysis, chromatography, spectral data, etc.).

The methods described in this section, namely PARAFAC and NPLS, are generalization of the PCA and PLS approaches, discussed in Section 4.5. They provide the multi-modal data treatment. Unlike the vector oriented algorithms that can be applied to the multi-way (tensor) data after unfolding, the multi-way models preserve the structure of the data, improve robustness of the results as well as allows identifying relative impact of each feature.

In this section explanatory and response variables are represented by the N and M-order tensors, respectively: $\underline{\mathbf{x}} \in \mathbb{R}^{I_1 \times \ldots \times I_N}$ and $\underline{\mathbf{y}} \in \mathbb{R}^{I_1 \times \ldots \times I_M}$. After observation of n data samples of both types of variables, tensors are formed by means of concatenation of the data samples along the first modality: $\underline{\mathbf{X}} \in \mathbb{R}^{n \times I_1 \times \ldots \times I_N}$ and $\underline{\mathbf{Y}} \in \mathbb{R}^{n \times I_1 \times \ldots \times I_M}$.

4.6.1 PARAFAC

PARAFAC (Parallel Factor analysis) is a generalization of the PCA approach to data structures of high orders (Harshman, 1970). It is used for decomposition of the tensor into sets of score and loading vectors, that describes the data in a compressed way.

The PARAFAC decomposition of the observation tensor $\underline{\mathbf{X}} \in \mathbb{R}^{n \times I_1 \times \ldots \times I_N}$ is expressed in the form of outer product (Bro, 1996):

$$\underline{\mathbf{X}} = \sum_{f=1}^{F} \mathbf{t}_f \circ \mathbf{w}_f^1 \circ \ldots \circ \mathbf{w}_f^N + \underline{\mathbf{E}}, \qquad \left\|\underline{\mathbf{E}}\right\|_F \to \min \tag{4.14}$$

where, $\mathbf{T} = [\mathbf{t}_1 | \ldots | \mathbf{t}_F]$ is a matrix of the $F \leq n$ extracted score vectors, $\underline{\mathbf{W}}^i = [\mathbf{w}_1^i | \ldots | \mathbf{w}_F^i]$ represents a matrix of the extracted loadings for i-th modality ($i = 1, \ldots, N$), and $\underline{\mathbf{E}}$ is a tensor of residuals. The tensor $\underline{\mathbf{X}}$ assumed to be mean-centered along its first modality. As in the case of PCA, the loadings are chosen in a such way to minimize the norm of the residual tensor $\left\|\underline{\mathbf{E}}\right\|_F$. Thus, PARAFAC provides the best approximation of the tensor $\underline{\mathbf{X}}$ by the sum of F rank-one tensors.

To find the solution of the problem (4.14) the Alternating Least Squares (ALS) algorithm (Yates, 1933) could be applied (Bro, 1996). ALS fixes all parameters except one, which is estimated in a least square sense. The procedure is repeated for all parameters until convergence. Unfortunately, the ALS algorithm, as well as its simple modifications, suffers from unstable convergence properties, moreover, the solution is quite sensible with respect to noise (Bro, 1998; Cichocki et al., 2009; Albright et al., 2006). However, the algorithm is often utilized because of its programming simplicity (Comon, 2002).

To explain PARAFAC and ALS algorithms in details, let us consider the case of third-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{n \times I_1 \times I_2}$.

Algorithm 4.1 PARAFAC

Input: $\underline{\mathbf{X}}$, number of factors F.

Outputs: $\left\{\mathbf{t}_f, \mathbf{w}_f^1, \mathbf{w}_f^2\right\}_{f=1}^{F}$.

1. $\underline{\mathbf{X}}_0 = \underline{\mathbf{X}}$
2. **for** $f = 1$ to F
3. $\{\mathbf{t}_f, \mathbf{w}_f^1, \mathbf{w}_f^2\} = \text{ALS}(\underline{\mathbf{X}}_{f-1})$
4. $\underline{\mathbf{X}}_f = \underline{\mathbf{X}}_{f-1} - \mathbf{t}_f \circ \mathbf{w}_f^1 \circ \mathbf{w}_f^2$ // *deflation*
5. **end for**

[a] The vector outer product is denoted as $\mathbf{w} \circ \mathbf{v}$ (see Section 4.2.1).

Algorithm 4.2 Alternating Least Square (ALS)

Input: $\underline{\mathbf{A}}$.

Outputs: $\mathbf{a}^1$, $\mathbf{a}^2$, $\mathbf{a}^3$.

1. Initialization of $\mathbf{a}^1$, $\mathbf{a}^2$, $\mathbf{a}^3$.
2. **while** not converge $\mathbf{a}^1$, $\mathbf{a}^2$, and $\mathbf{a}^3$ **do**
3. $\mathbf{B}_1 = \text{vec}(\mathbf{a}^2(\mathbf{a}^3)^T)$
4. $\mathbf{a}^1 = \underline{\mathbf{A}}_{(1)}\mathbf{B}_1(\mathbf{B}_1^T\mathbf{B}_1)^+$ // *least square solution*
5. $\mathbf{B}_2 = \text{vec}(\mathbf{a}^1(\mathbf{a}^3)^T)$
6. $\mathbf{a}^2 = \underline{\mathbf{A}}_{(2)}\mathbf{B}_2(\mathbf{B}_2^T\mathbf{B}_2)^+$ // *least square solution*
7. $\mathbf{B}_3 = \text{vec}(\mathbf{a}^1(\mathbf{a}^2)^T)$
8. $\mathbf{a}^3 = \underline{\mathbf{A}}_{(3)}\mathbf{B}_3(\mathbf{B}_3^T\mathbf{B}_3)^+$ // *least square solution*
9. **end while**

[a] $\text{vec}(\mathbf{A})$ is vectorization of a matrix $\mathbf{A}$ (see Section 0).

[b] $\underline{\mathbf{A}}_{(i)}$ is the mode-i unfolding of the tensor $\underline{\mathbf{A}}$ (see Section 0).

[c] $\mathbf{A}^+$ is the Moore–Penrose pseudoinverse of $\mathbf{A}$ (Golub and Van Loan, 1996).

Let us note that the PARAFAC solution is unique up to an arbitrary factor (Bro, 1996). Thus, PARAFAC has no rotation freedom unlike PCA, and, therefore, no post-processing is needed to detect the most appropriate solution.

Graphical representation of the PARAFAC approach is shown in Figure 4.8.

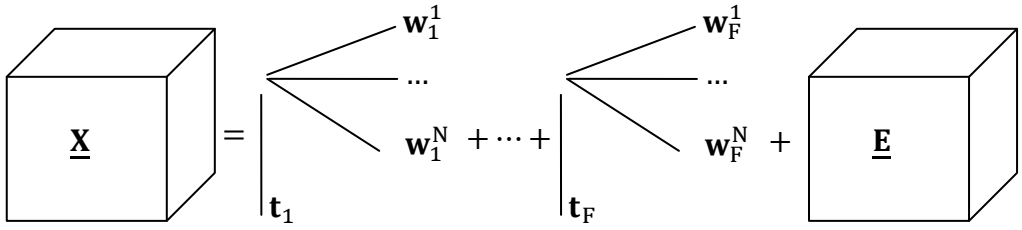

Figure 4.8 Graphical representation of the PARAFAC decomposition of a tensor $\underline{\mathbf{X}}$.

After decomposition of the tensor $\underline{\mathrm{X}}$, a regression of the response tensor $\underline{\mathbf{Y}}$ on the matrix $\mathbf{T}$ is build. For simplicity, we will consider the vector case of response variable: $\mathbf{y} \in \mathbb{R}^n$. Then, for example, a linear regression could be constructed $\hat{\mathbf{y}} = \mathbf{T}\hat{\mathbf{b}}$, $\hat{\mathbf{b}} = (\mathbf{T}^T\mathbf{T})^{-1}\mathbf{T}^T\mathbf{y}$.

4.6.2 Multi-linear PLS Regression (NPLS)

Multi-linear or N-way PLS Regression (NPLS) was introduced as generalization of the two-way PLS to data of higher orders (Bro, 1996). The goal of this method is the model the relation between the explanatory tensor $\underline{\mathbf{X}} \in \mathbb{R}^{n \times I_1 \times \ldots \times I_N}$ and the tensor of responses $\underline{\mathbf{Y}} \in \mathbb{R}^{n \times I_1 \times \ldots \times I_M}$.

The NPLS decomposition of the observation tensor $\underline{\mathbf{X}} \in \mathbb{R}^{n \times I_1 \times \ldots \times I_N}$ and the response tensor $\underline{\mathbf{Y}} \in \mathbb{R}^{n \times I_1 \times \ldots \times I_M}$ is expressed as:

$$\begin{aligned} \underline{\mathbf{X}} &= \sum_{f=1}^{F} \mathbf{t}_f \circ \mathbf{w}_f^1 \circ \ldots \circ \mathbf{w}_f^N + \underline{\mathbf{E}}, \\ \underline{\mathbf{Y}} &= \sum_{f=1}^{F} \mathbf{u}_f \circ \mathbf{q}_f^1 \circ \ldots \circ \mathbf{q}_f^M + \underline{\mathbf{F}}, \\ \mathrm{cov}(\mathbf{t}_f, \mathbf{u}_f) &\to \max, f = 1, \ldots, F \end{aligned} \tag{4.15}$$

General scheme of NPLS is depicted in Figure 4.9.

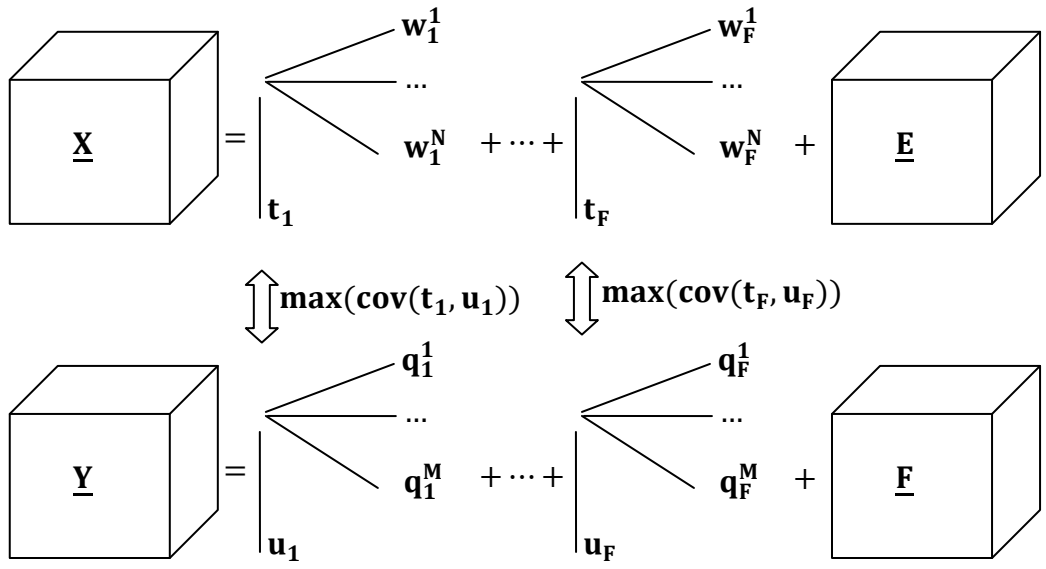

Figure 4.9 Graphical representation of the NPLS decomposition of the tensors $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$.

Let us consider the NPLS algorithm in the particular case of a fourth-order tensor of observations $\underline{\mathbf{X}} \in \mathbb{R}^{n \times I_1 \times I_2 \times I_3}$ and a vector of observations of the dependent variable $\mathbf{y} \in \mathbb{R}^n$. This case was chosen since it corresponds to our BCI experiments. A set of latent variables $\mathbf{t}_f \in \mathbb{R}^n$ is extracted consequently from the first mode of $\underline{\mathbf{X}}$. It provides maximum covariance between $\mathbf{t}_f$ and $\mathbf{y}_f$ ($\mathbf{y}_f = \mathbf{u}_f$) at the iteration f. In parallel, by means of the ALS (Algorithm 4.2) a set of vectors $\{\mathbf{w}^1 \in \mathbb{R}^{I_1}, \mathbf{w}^2 \in \mathbb{R}^{I_2}, \mathbf{w}^3 \in \mathbb{R}^{I_3}\}_f$, related to the second, the third, and the fourth modes of $\underline{\mathbf{X}}$, respectively , is formed in such a way that projection of the tensor $\underline{\mathbf{X}}$ on these vectors results in $\mathbf{t}_f$. Coefficients $\mathbf{b}_f$ of the linear regression of $\mathbf{y}$, depending on the current set of latent variables $\{\mathbf{t}_i\}_{i=1}^{f}$, are calculated. This regression is used for $\mathbf{y}$ deflation on each iteration. Besides that, the tensor $\underline{\mathbf{X}}$ is also deflated according to the decomposition. Let us note that there are modifications of NPLS (e.g., de Jong, 1998), where deflation of the tensor $\underline{\mathbf{X}}$ is not applied. The procedure is repeated a pre-defined number F of times. In details, the pseudo-code of the method is presented by Algorithm 4.3.

Algorithm 4.3 NPLS

Input: $\underline{\mathbf{X}}$, $\mathbf{y}$, number of factors F.

Outputs: $\{\mathbf{w}_f^1, \mathbf{w}_f^2, \mathbf{w}_f^3\}_{f=1}^F, \{\mathbf{b}_f\}_{f=1}^F$.

1. Centering of $\{\underline{\mathbf{X}},\ \mathbf{y}\}$
2. $\mathbf{y}_1 = \mathbf{y}$
3. $\underline{\mathbf{X}}_1 = \underline{\mathbf{X}}$
4. **for** $f = 1$ to F
5. $\underline{\mathbf{Z}}_f = \underline{\mathbf{X}}_f \times_1 \mathbf{y}_f$
6. $\{\mathbf{w}_f^1, \mathbf{w}_f^2, \mathbf{w}_f^3\}$ = ALS$(\underline{\mathbf{Z}}_f)$
7. $\mathbf{w}_f^m = \mathbf{w}_f^m / \|\mathbf{w}_f^m\|, \quad m = 1,2,3$
8. $\mathbf{t}_f = \underline{\mathbf{X}}_f \times_4 \mathbf{w}_f^3 \times_3 \mathbf{w}_f^2 \times_2 \mathbf{w}_f^1$
9. $\mathbf{T}_f = [\mathbf{t}_1 | \ldots | \mathbf{t}_f]$
10. $\mathbf{b}_f = (\mathbf{T}_f{}^T \mathbf{T}_f)^{-1} \mathbf{T}_f{}^T \mathbf{y}_f$
11. $\mathbf{y}_{f+1} = \mathbf{y}_f - \mathbf{T}_f \mathbf{b}_f$
12. $\underline{\mathbf{X}}_{f+1} = \underline{\mathbf{X}}_f - \mathbf{t}_f \circ \mathbf{w}_f^1 \circ \mathbf{w}_f^2 \circ \mathbf{w}_f^3$ // *deflation*
13. **end for**

[a] The n-mode vector product of a tensor $\underline{\mathbf{X}}$ and vector $\mathbf{v} \in \mathbb{R}^{I_n}$ is denoted as $\underline{\mathbf{X}} \times_n \mathbf{v}$ (see Section 0).

[b] The vector outer product is denoted as $\mathbf{w} \circ \mathbf{v}$ (see Section 4.2.1).

Sets of the regression coefficients $\{\mathbf{b}_f\}_{f=1}^F$ and the projection vectors $\{\mathbf{w}_f^1, \mathbf{w}_f^2, \mathbf{w}_f^3\}_{f=1}^F$ formed by Algorithm 4.3 are used for prediction of the output variable $\hat{\mathbf{y}}^{new}$ for the new data.

In details, the prediction steps are shown in Algorithm 4.4.

Algorithm 4.4 NPLS Prediction

Input: $\underline{\mathbf{X}}^{new} \in \mathbb{R}^{k \times I_1 \times I_2 \times I_3}, \{\mathbf{w}_f^1, \mathbf{w}_f^2, \mathbf{w}_f^3\}_{f=1}^F, \{\mathbf{b}_f\}_{f=1}^F$.

Outputs: $\hat{\mathbf{y}}^{new} \in \mathbb{R}^k$.

1. Centering of $\underline{\mathbf{X}}^{new}$ in the same way as on the training set
2. $\underline{\mathbf{X}}_1^{new} = \underline{\mathbf{X}}^{new}$
3. **for** $f = 1$ to F

4. $\mathbf{t}_f = \underline{\mathbf{X}}_f^{new} \times_4 \mathbf{w}_f^3 \times_3 \mathbf{w}_f^2 \times_2 \mathbf{w}_f^1$
5. $\mathbf{T}_f = [\mathbf{t}_1 | \ldots | \mathbf{t}_f]$
6. $\underline{\mathbf{X}}_{f+1}^{new} = \underline{\mathbf{X}}_f^{new} - \mathbf{t}_f \circ \mathbf{w}_f^1 \circ \mathbf{w}_f^2 \circ \mathbf{w}_f^3$ // *deflation*
7. **end for**
8. $\hat{\mathbf{y}}^{new} = \sum_{f=1}^{F} \mathbf{T}_f \mathbf{b}_f$
9. De-centering of $\hat{\mathbf{y}}^{new}$

[a] The n-mode vector product of a tensor $\underline{\mathbf{X}}$ and vector $\mathbf{v} \in \mathbb{R}^{I_n}$ is denoted as $\underline{\mathbf{X}} \times_n \mathbf{v}$ (see Section 4.2.1).

[b] The vector outer product is denoted as $\mathbf{w} \circ \mathbf{v}$ (see Section 4.2.1).

Both PARAFAC and NPLS are widespread projective methods for the case of highly correlated tensor data of high dimensionality. However, contrary to PARAFAC, NPLS constructs a set of projectors taking into account information about the response variables. Thus, it was chosen as the basis approach for our BCI experiments.

4.6.3 Modality Influence Analysis

The elements of the input data have an implicit impact on the NPLS regression model through the latent variables. The Modality Influence (MI) analysis (Cook and Weisberg, 1982; Martens and Næs, 1989; Bro, 1996; Nilsson et al, 1997) allows estimating the relative importance of the elements of each mode for the final model. For example, in BCI experiments MI analysis can be applied to estimate the importance of electrodes, frequency bands, and time intervals (Eliseyev et al., 2011a).

Estimation $\hat{\mathbf{y}}$ of the dependent variable $\mathbf{y}$ according to the model $\hat{\mathbf{y}} = \sum_{f=1}^{F} \mathbf{T}_f \mathbf{b}_f$ can be represented as:

$$\hat{\mathbf{y}} = \sum_{f=1}^{F} \mathbf{T}_f \mathbf{b}_f = \sum_{f=1}^{F} \alpha_f \mathbf{t}_f = \sum_{f=1}^{F} (\alpha_f \|\mathbf{t}_f\|) \frac{\mathbf{t}_f}{\|\mathbf{t}_f\|}$$

$$= \sum_{f=1}^{F} \alpha_f^* \frac{\mathbf{t}_f}{\|\mathbf{t}_f\|}.$$

For the given mode i, which is characterized by the set of the projectors $\mathbf{w}^i = \{\mathbf{w}_f^i\}_{f=1}^F$, the matrix $\mathbf{A}^i = [\alpha_1^* \mathbf{w}_1^i | \ldots | \alpha_F^* \mathbf{w}_F^i]$ is formed. The vector of leverages for the i-th modality is defined as $\mathbf{h}^i = diag\left(\mathbf{A}^i (\mathbf{A}^i)^T\right)$. All the elements of the vector $\mathbf{h}^i$ are

non-negative. Moreover, the elements of the leverage which are close to zero have not affected the model very much, while the elements with high values are more important. This information could be useful for reduction of the modality dimension. Those components of the modality which have relatively small weights in the vector of leverages can be excluded from the further analysis.

4.6.4 Shortcomings of the generic NPLS approach

Despite many positive properties, the generic NPLS method has drawbacks. For instance, the identified model is of excessive complexity. Other shortcoming of the generic approach consists in its significant consumption of memory resources. The method continuously keeps in the active memory the tensor $\underline{\mathbf{X}}$, size of which could be too great. Finally, generic NPLS cannot be used online for adaptive adjustment of the identified model to changings in the data flow.

In this study three modifications of the generic NPLS approach are proposed. A penalized version of the generic algorithm, providing an opportunity for selection of informative features, is represented in Section 4.7. An iterative version of the generic algorithm, which allows storing only part of the tensor $\underline{\mathbf{X}}$ in the active memory, is described in Section 4.8. Presented in Section 4.9 a recursive NPLS algorithm provides an opportunity of online treatment of the data.

4.7 L1-Penalized N-PLS algorithm (PNPLS)

In this study we propose generalization of the NPLS algorithm which gives us an opportunity to impose the additional constrains on feature selection. For this purpose, the Alternating Least Squares (Algorithm 4.2) can be substituted for its penalized version and used for decomposition of the tensor $\underline{\mathbf{Z}} = \underline{\mathbf{X}} \times_1 \mathbf{y}$ in the NPLS algorithm (see Algorithm 4.3; Eliseyev et al., 2012). In the general case, the optimization problem for the Penalized ALS algorithm has the form:

$$\{\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2, \hat{\mathbf{a}}_3\} = \underset{\mathbf{a}_1,\mathbf{a}_2,\mathbf{a}_3}{\operatorname{argmin}} \left(\left\| \underline{\mathbf{A}} - \mathbf{a}_1 \circ \mathbf{a}_2 \circ \mathbf{a}_3 \right\|_F^2 + \lambda P(\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3) \right) \tag{4.16}$$

here, $\underline{\mathbf{A}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, $\mathbf{a}_i \in \mathbb{R}^{I_i}$, $i = 1,2,3$, $P(\cdot)$ is a penalization term, and $\lambda \in \mathbb{R}^+$ is a nonnegative regularization parameter, which quantifies the relative importance of the penalization. Depending on the penalization operator $P(\cdot)$, several formulations of the optimization task can be considered. Detailed description of different optimization

problems could be found in the work of Martínez-Montes et al. (2008). One particular case is the well-known Ridge regression, where the penalty function is $P_{RR}(\mathbf{A}) = \|\mathbf{L}_1\mathbf{A}\mathbf{L}_2\|^2$, with $\mathbf{L}_1$ and $\mathbf{L}_2$ being two operators that operate on the columns and rows of $\mathbf{A}$, respectively (see, Hoerl and Kennard (2000)). However, to have a sparse solution $P(\mathbf{A})$ could be chosen as a non-convex function with a singularity at the origin. For example, ℓ_1-norm penalty can be often used: $\|\mathbf{A}\|_1 = \sum_{i,j}|a_{ij}|$. Some particular cases of the function $P(\mathbf{A})$ are:

- $P_{LASSO}(\mathbf{A}) = \|\mathbf{A}\|_1$, the Least Absolute Shrinkage Selection Operator (LASSO) (Tibshirani, 1996);
- $P_{FL}(\mathbf{A}) = \|\mathbf{L}\mathbf{A}\|_1$, the "Fusion Lasso" (Land and Friedman, 1996). Here, $\mathbf{L}$ is the first order difference operator;
- the Smooth Clipped Absolute Deviation (SCAD) (Fan, 1997);
- $P_{Enet}(\mathbf{A}) = \mu_1\|\mathbf{L}_1\mathbf{A}\|_1 + \mu_2\|\mathbf{L}_2\mathbf{A}\|^2$, the Elastic Net (Enet) (Zou and Hastie, 2005).

To obtain a sparse solution, the ℓ_1-norm penalty (LASSO) is often used. The LASSO can be easily implemented providing a sufficient level of selectivity. In the present study, the ℓ_1-penalty was integrated into the ALS (Algorithm 4.2). At each step of the ALS algorithm, all the projectors are fixed except one leading to the optimization:

$$\hat{\mathbf{a}}_i = \underset{\mathbf{a}_i}{\operatorname{argmin}} \left(\left\| \underline{\mathbf{A}} - \mathbf{a}_1 \circ \mathbf{a}_2 \circ \mathbf{a}_3 \right\|_F^2 + \lambda \|\mathbf{a}_i\|_1 \right), \qquad i = 1,2,3. \tag{4.17}$$

For the particular case $i = 1$:

$$\hat{\mathbf{a}}_1 = \underset{\mathbf{a}_1}{\operatorname{argmin}} \left(\left\| \underline{\mathbf{A}} - \mathbf{a}_1 \circ \mathbf{a}_2 \circ \mathbf{a}_3 \right\|_F^2 + \lambda \|\mathbf{a}_1\|_1 \right). \tag{4.18}$$

The optimization problem (4.18) can be rewritten in the matrix notation:

$$\hat{\mathbf{a}}_1 = \underset{\mathbf{a}_1}{\operatorname{argmin}} \left(\left\| \mathbf{A} - \mathbf{a}_1 \mathbf{a}_{2,3}^T \right\|_F^2 + \lambda \|\mathbf{a}_1\|_1 \right), \tag{4.19}$$

where $\mathbf{A} = \underline{\mathbf{A}}_{(1)}$ is unfolding of the tensor $\underline{\mathbf{A}}$, and $\mathbf{a}_{2,3} = \operatorname{vec}(\mathbf{a}_2 \circ \mathbf{a}_3)$ (see Section 0).

A possible approach to solve this optimization problem with the ℓ_1-penalization is the Gauss-Seidel algorithm (Shevade and Keerthi, 2003; Schmidt, 2005). The advantages of this algorithm are its simplicity and low computational cost, as well as low memory consumption which is of great importance for BCI tasks. We have applied this approach to solve the optimization task (4.19). Namely, the anti-gradient of the

penalized residuals squares sum function $PRSS = \left\|\mathbf{A} - \mathbf{a}_1\mathbf{a}_{2,3}^T\right\|_F^2 + \lambda\|\mathbf{a}_1\|_1$ was considered: $-G(\mathbf{a}_1) = 2\mathbf{a}_{2,3}^T\left(\mathbf{A}^T - \mathbf{a}_{2,3}\mathbf{a}_1^T\right) - \lambda sign(\mathbf{a}_1)$, where $\mathbf{v} = sign(\mathbf{a})$ is a vector with elements $v_i = +1$, if $a_i \geq 0$, and $v_i = -1$, otherwise. For the first iteration, $\mathbf{a}_1$ is set equal to zero consequently the anti-gradient $-G(\mathbf{0}) = 2\mathbf{a}_{2,3}^T\mathbf{A}^T - \lambda\mathbf{1}$. Then, the element of $\mathbf{a}_1$ with the largest magnitude of the anti-gradient is added to a set of 'free' variables. These 'free' variables are optimized in a 'one at a time' way. For details see Shevade and Keerthi, 2003. Note, that if $\lambda \geq \lambda_{max} = \max\left(2\mathbf{a}_{2,3}^T\mathbf{A}^T\right)$, the method returns as a solution $\hat{\mathbf{a}}_1 = \mathbf{0}$.

To automatically select the optimal value of λ different approaches can be used: Cross-validation (Devijver and Kittler, 1982), generalized cross-validation (Golub et al., 1979), Akaike's Information Criterion (Akaike, 1974), or Schwartz's Bayesian Information Criterion (Schwartz, 1978).

Penalized decomposition of the tensor $\underline{\mathbf{Z}} = \underline{\mathbf{X}} \times_1 \mathbf{y}$ results in a set of sparse factor $\{\mathbf{w}^1, \mathbf{w}^2, \mathbf{w}^3\} = \{\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2, \hat{\mathbf{a}}_3\}$ (see Algorithm 4.3).

Thus, in the work the task of Penalized NPLS algorithm is formulated. For its solving the Alternative Penalized Least Squares is proposed.

The L1-Penalized NPLS algorithm combines computational simplicity and moderate memory consumption with sufficient selectivity. Thus, it could be applied for BCI system calibration and for electrode subset selection.

4.8 Iterative N-PLS algorithm (INPLS)

In spite of efficiency, the curse-of-dimensionality has significant influence on the application of NPLS to the BCI systems. The amount of data increases exponentially with dimensionality of the feature tensor/vector. Thus, restrictions on the data set size considerably limits frequency/temporal resolution of the signal decomposition, as well as the number of simultaneously analyzed electrodes, due to huge storage consumption for saving the tensor $\underline{\mathbf{X}}$ in the active memory. To overcome the problem, we proposed the Iterative Multi-way Partial Least Squares (INPLS) algorithms (Eliseyev et al., 2011a), which properly treat the data set of huge dimension. It is based on fragmentation of the initial dataset on several subsets and their sequential treatment. Thus, at each instant only a small part of the data is stored in the active

memory. The algorithm was tested with artificial and real datasets and demonstrated efficiency and robustness (see Sections 4.8.2 for artificial and Section 6.1 for real datasets).

4.8.1 Algorithm description

The main steps of the INPLS are shown in Algorithm 4.5. The training dataset is preliminary segmented on L subsets $\left\{\underline{\mathbf{X}}^{(l)} \in \mathbb{R}^{n_l \times I_1 \times I_2 \times I_3},\ \mathbf{y}^{(l)} \in \mathbb{R}^{n_l}\right\}$. A set of three projection vectors $\{\mathbf{w}^1, \mathbf{w}^2, \mathbf{w}^3\}$ (according to the number of modalities) is constructed by successive refinement of their values on every subset $\left\{\underline{\mathbf{X}}^{(l)},\ \mathbf{y}^{(l)}\right\}$. For this purpose the iterative Tensor Factorization is used (see Algorithm 4.6). It is again based on the ALS algorithm (see Algorithm 4.2) in which previously calculated tensor decomposition is taken as initial estimation. The current approximation of the projection vectors is obtained as a weighted sum of the weights generated by ALS on the current iteration and the vectors from the previous iteration. After the set of projectors $\{\mathbf{w}^1, \mathbf{w}^2, \mathbf{w}^3\}$ is identified, the generic NPLS algorithm is applied for determination of the regression coefficients in the space of the latent variables (see Algorithm 4.3: steps 8-12). This procedure is repeated a pre-defined number F of times. The set of coefficients vectors $\left\{\mathbf{b}_f\right\}_{f=1}^{F}$ and projection vectors $\left\{\mathbf{w}_f^1, \mathbf{w}_f^2, \mathbf{w}_f^3\right\}_{f=1}^{F}$ are used for prediction in the same way as in traditional NPLS (Algorithm 4.4). Graphical representation of the algorithm is shown in Figure 4.10.

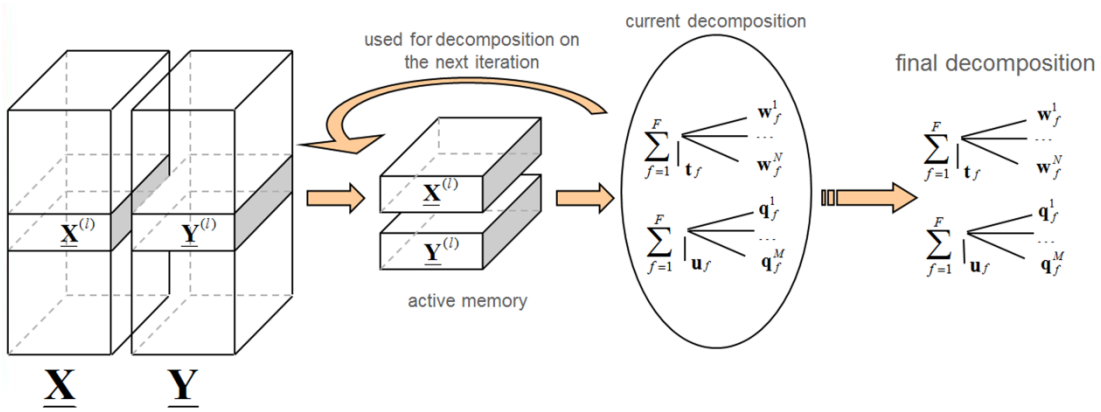

Figure 4.10 Graphical representation of the INPLS algorithm applied to the tensors $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$ (the general case: $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$ are the tensors of the orders N and M, respectively).

Algorithm 4.5 Iterative NPLS

Input: $\left\{\underline{\mathbf{X}}^{(l)},\ \mathbf{y}^{(l)}\right\}_{l=1}^{L}$, number of factors F.

Outputs: $\left\{\mathbf{w}_f^1, \mathbf{w}_f^2, \mathbf{w}_f^3\right\}_{f=1}^{F}, \left\{\mathbf{b}_f\right\}_{f=1}^{F}$.

1. Centering of $\left\{\underline{\mathbf{X}},\ \mathbf{y}\right\}$
2. $\mathbf{y}_1 = \mathbf{y}$
3. $\underline{\mathbf{X}}_1 = \underline{\mathbf{X}}$
4. **for** $f = 1$ to F
5. $\left\{\mathbf{w}_f^1, \mathbf{w}_f^2, \mathbf{w}_f^3\right\} = \left\{\mathbf{1}^{I_1}/\|\mathbf{1}^{I_1}\|,\ \mathbf{1}^{I_2}/\|\mathbf{1}^{I_2}\|,\ \mathbf{1}^{I_3}/\|\mathbf{1}^{I_3}\|\right\}$
6. **for** $l = 1$ **to** L
7. $\underline{\mathbf{Z}}_f^{(l)} = \underline{\mathbf{X}}_f^{(l)} \times_1 \mathbf{y}_f^{(l)}$
8. $\left\{\mathbf{w}_f^1, \mathbf{w}_f^2, \mathbf{w}_f^3\right\}$ = Tensor Factorization $\left(\underline{\mathbf{Z}}_f^{(l)},\ \left\{\mathbf{w}_f^1, \mathbf{w}_f^2, \mathbf{w}_f^3\right\},\ l\right)$
9. **end for**
10. $\mathbf{w}_f^m = \mathbf{w}_f^m / \left\|\mathbf{w}_f^m\right\|, \quad m = 1,2,3$
11. $\mathbf{t}_f = \underline{\mathbf{X}}_f \times_4 \mathbf{w}_f^3 \times_3 \mathbf{w}_f^2 \times_2 \mathbf{w}_f^1$
12. $\mathbf{T}_f = \left[\mathbf{t}_1 | \ldots | \mathbf{t}_f\right]$
13. $\mathbf{b}_f = \left(\mathbf{T}_f{}^T \mathbf{T}_f\right)^{-1} \mathbf{T}_f{}^T \mathbf{y}_f$
14. $\mathbf{y}_{f+1} = \mathbf{y}_f - \mathbf{T}_f \mathbf{b}_f$
15. $\underline{\mathbf{X}}_{f+1} = \underline{\mathbf{X}}_f - \mathbf{t}_f \circ \mathbf{w}_f^1 \circ \mathbf{w}_f^2 \circ \mathbf{w}_f^3$
16. **end for**

[a] The n-mode vector product of a tensor $\underline{\mathbf{X}}$ and vector $\mathbf{v} \in \mathbb{R}^{I_n}$ is denoted as $\underline{\mathbf{X}} \times_n \mathbf{v}$ (see Section 0).

[b] The vector outer product is denoted as $\mathbf{w} \circ \mathbf{v}$ (see Section 4.2.1).

Algorithm 4.6 Tensor Factorization

Input: $\underline{\mathbf{Z}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, $\{\mathbf{w}^1, \mathbf{w}^2, \mathbf{w}^3\}$, l.

Outputs: $\{\mathbf{w}_{new}^1, \mathbf{w}_{new}^2, \mathbf{w}_{new}^3\}$.

1. $\left\{\mathbf{w}_{tmp}^1, \mathbf{w}_{tmp}^2, \mathbf{w}_{tmp}^3\right\}$ = ALS$\left(\underline{\mathbf{Z}},\ \{\mathbf{w}^1, \mathbf{w}^2, \mathbf{w}^3\}\right)$

2. $\mathbf{w}_{new}^{m} = (\mathbf{w}_{new}^{m} + (l-1)\mathbf{w}^{m})/l, \quad m = 1,2,3$

[a] ALS$\left(\underline{\mathbf{Z}}, \{\mathbf{w}^1, \mathbf{w}^2, \mathbf{w}^3\}\right)$ means computation of ALS (see Algorithm 4.2) for factorization of the tensor $\underline{\mathbf{Z}}$ from the initial approximation $\{\mathbf{w}^1, \mathbf{w}^2, \mathbf{w}^3\}$.

4.8.2 Test of algorithm with simulated data. Comparison INPLS with generic NPLS

This test intends to study performance and prediction accuracy of INPLS in different conditions including the noise level and to compare the iterative algorithm (INPLS) with traditional one (NPLS). Taking into account the task specificity, i.e., 'event'–'non-event' detection, the test was performed for binary output variables. An artificial data set $\left\{\underline{\mathbf{x}}_k \in \mathbb{R}^{100\times200},\ y_k \in \{0,\ 1\}\right\}_{k=1}^{1100}$ was formed in the following way. Binary $y_k \in \{0,\ 1\}$ was randomly generated with equal probabilities. Tensors $\underline{\mathbf{x}}_k$ were calculated according to the rule: $\underline{\mathbf{x}}_k = \underline{\mathbf{c}}(y_k) + \lambda\underline{\varepsilon}_k$, where $\underline{\mathbf{c}}(y_k) \in \mathbb{R}^{100\times200}$ is set as $c_{ij} = \cos\left((i - j/2)\pi/2 + 2\right)$ if $y_k = 0$, and $c_{ij} = \sin\left((i + j/2)\pi/2 + 1\right)$ if $y_k = 1$. The random noise $\underline{\varepsilon}_k \in R^{100\times200}$ was drawn from a multivariate normal distribution $\mathcal{N}\left(\underline{\mathbf{0}}, \underline{\Sigma}\right)$, which is a generalization of Gaussian distribution for tensors (Hamedani and Tata, 1975). Parameter λ is introduced to control the signal-to-noise ratio (SNR). The noise has the same amplitude as the signal $\underline{\mathbf{c}}(y_k)$ in the case of $\lambda = 1$. Figure 4.11 shows the examples of graphical representation of tensors $\underline{\mathbf{x}}$ for different values of λ.

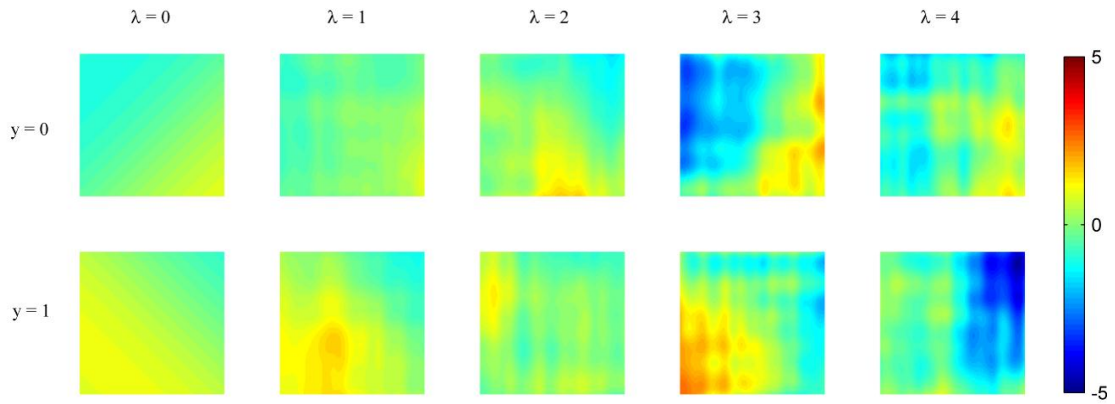

Figure 4.11 Example of points $\underline{\mathbf{x}}_k$, for $y_k = 0,\ 1$ with different noise levels: $\lambda = 0,\ 1,\ 2,\ 3,\ 4$ presenting the artificial dataset used for INPLS performance study.

Generated data points were split into the training and the test datasets with 100 points $\{\underline{\mathbf{x}}_k,\ y_k\}_{k=1}^{100}$ and 1000 points $\{\underline{\mathbf{x}}_k^{test},\ y_k^{test}\}_{k=1}^{1000}$, respectively. From these datasets the training and the test tensors $\{\underline{\mathbf{X}} \in \mathbb{R}^{100\times100\times200},\ \mathbf{y} \in \{0,\ 1\}^{100}\}$, $\{\underline{\mathbf{X}}^{test} \in \mathbb{R}^{1000\times100\times200},\ \mathbf{y}^{test} \in \{0,\ 1\}^{1000}\}$ were formed.

A prediction accuracy of the iterative algorithm was compared to NPLS depending on the training data fragmentation manner (i.e., the number of points in the subsets), the level of the noise, and the number of factors used by the algorithms. The training set was split in four ways $S5$, $S10$, $S20$, and $S100$ with 5, 10, 20 and 100 points per subset, respectively:

$$S5 = \{\underline{\mathbf{X}}^{(l)} \in \mathbb{R}^{5\times100\times200},\ \mathbf{y}^{(l)} \in \{0,\ 1\}^{5}\}_{l=1}^{20},$$

$$S10 = \{\underline{\mathbf{X}}^{(l)} \in \mathbb{R}^{10\times100\times200},\ \mathbf{y}^{(l)} \in \{0,\ 1\}^{10}\}_{l=1}^{10},$$

$$S20 = \{\underline{\mathbf{X}}^{(l)} \in \mathbb{R}^{20\times100\times200},\ \mathbf{y}^{(l)} \in \{0,\ 1\}^{20}\}_{l=1}^{5},$$

$$S100 = \{\underline{\mathbf{X}}^{(l)} \in \mathbb{R}^{100\times100\times200},\ \mathbf{y}^{(l)} \in \{0,\ 1\}^{100}\}_{l=1}^{1}.$$

Here, $S100$ corresponds to traditional NPLS. The number of factors F equal to 2, 5, and 10 was considered. The noise amplitude λ varied from 1 to 4, i.e., from 100% to 400% of noise. Resulted regression models were validated on the test dataset. General performance of the algorithm was estimated by comparison of predicted $\hat{y}_k$ with y_k using Euclidian distance: $(\sum_{k=1}^{1000}|y_k - \hat{y}_k|^2)^{0.5}/1000$.

In all the tests, INPLS either demonstrated comparable results or surpassed traditional NPLS. Moreover, INPLS noticeably outperformed NPLS in the case of large factor number and high noise level($F = 10, \lambda \geq 3$) (Figure 4.12). This advantage can be explained by overfitting effect suppression in INPLS. Proposed iterative algorithm demonstrated good accuracy and was applied for the BCI system calibration.

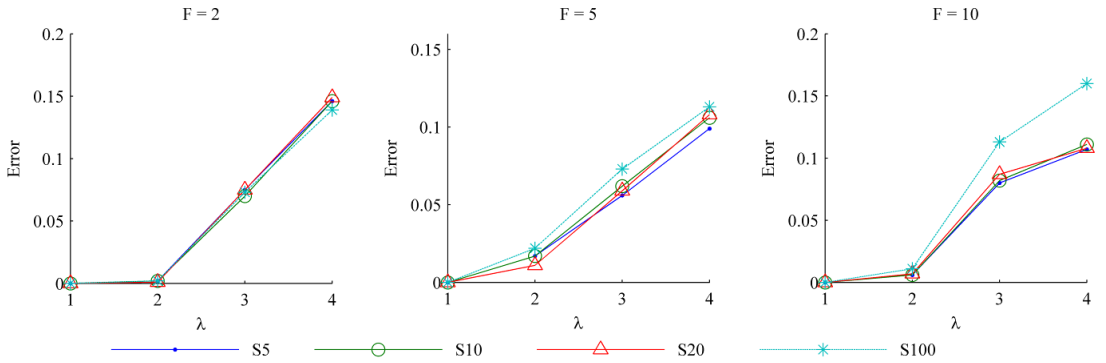

Figure 4.12 Comparison of INPLS vs. NPLS for the different noise level λ, the factor number F and data fragmentation manner $S5$, $S10$, $S20$, and $S100$. Here, $S100$ corresponds to generic NPLS.

4.9 Recursive N-PLS algorithm (RNPLS)

One of the major problems in BCI studies is the variability of the neuronal signals, in particular, due to the brain plasticity. These changes in the neuronal activity require recalibration of the BCI systems. The full system recalibration is a time and labor consuming procedure. Adaptive calibration aims to provide a fast adjustment of the BCI system to mild changes of the signal. Although INPLS allows treating data arrays of huge dimension, this method cannot be applied for adaptive learning. In this section the Recursive NPLS (RNPLS) algorithm is proposed (Eliseyev et al., 2011b). It allows online processing of the multi-modal data. Moreover, weighted RNPLS can be applied for adaptive learning to treat time-dependent recordings.

4.9.1 Algorithm description

The RNPLS algorithm unites the RPLS scheme of recursive calculation with multimodal data representation of NPLS. To apply the recursive approach to the NPLS algorithm described above, orthonormality of the latent variables as well as orthogonality of the latent variables to the residuals should be provided (see conditions (4.10), (4.11), and (4.12)).

Let us consider a fourth-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{n \times I_1 \times I_2 \times I_3}$ of observations and a vector of observations $\mathbf{y} \in \mathbb{R}^n$. At the first step, the tensor $\underline{\mathbf{X}}$ is represented by a set of factors $\{\mathbf{w}_1^i\}_{i=1}^3$:

$$\underline{\mathbf{X}} = \mathbf{t}_1 \circ \mathbf{w}_1^1 \circ \mathbf{w}_1^2 \circ \mathbf{w}_1^3 + \underline{\mathbf{E}}, \tag{4.20}$$

constructed by the NPLS algorithm (Algorithm 4.3). For simplicity of notation, let us unfold the tensor $\underline{\mathbf{X}}$ along the first mode into the matrix $\mathbf{X} \in \mathbb{R}^{n\times(I_1 I_2 I_3)}$ (Kolda and Bader, 2007). At the same time let us denote as $\mathbf{w_1} \in \mathbb{R}^{I_1 I_2 I_3}$ vectorization of a tensor $\underline{\mathbf{w}}_1 = \mathbf{w}_1^1 \circ \mathbf{w}_1^2 \circ \mathbf{w}_1^3$. The score vector $\mathbf{t}_1$ is a projection of the observation matrix $\mathbf{X}$ on $\mathbf{w}_1$: $\mathbf{t}_1 = \mathbf{X}\mathbf{w_1}$. In this matrix notation, Equation (4.20) can be rewritten in the form $\mathbf{X} = \mathbf{t}_1\mathbf{w}_1^T + \mathbf{E}_1$ (on iteration f: $\mathbf{X} = \mathbf{T}_f\mathbf{W}_f^T + \mathbf{E}_f$).

In general, the NPLS latent vector variables are not orthogonal, but this restriction is necessary for the recursive scheme of calculation (see Qin, 1998). Let us apply orthonormalization of the latent variables: $\mathbf{T}_f^{\perp} = \mathbf{T}_f\mathbf{A}_f$, $\left(\mathbf{T}_f^{\perp}\right)^T\mathbf{T}_f^{\perp} = \mathbf{I}_f$. Here, $\mathbf{A}_f$ is a matrix of orthonormalization. It could be obtained from the QR decomposition procedure (Golub and Van Loan, 1996), where any rectangular matrix $\mathbf{X} \in \mathbb{R}^{n\times m}, n \geq m$ could be represented as a product of two matrices $\mathbf{X} = \mathbf{QR}$ ($\mathbf{Q} \in \mathbb{R}^{n\times m}$: $\mathbf{Q}^T\mathbf{Q} = \boldsymbol{I}_m$, $\mathbf{R} \in \mathbb{R}^{m\times m}$ is an upper triangular matrix). There are several methods for computing the QR decomposition, such as the Gram–Schmidt process (Golub and Van Loan, 1996), Householder transformations (Householder, 1958), or Givens rotations (Givens, 1958), each has a number of advantages and disadvantages. For the new orthonormal latent variables the matrix $\mathbf{X}$ could be decomposed as $\mathbf{X} = \mathbf{T}_f\mathbf{W}_f^T + \mathbf{E}_f = \mathbf{T}_f^{\perp}\mathbf{P}_f^T + \mathbf{E}_f$, where $\mathbf{P}_f^T = \mathbf{A}_f^{-1}\mathbf{W}_f^T$. The Ordinary Least Squares (OLS) coefficients of regression of $\mathbf{y}_f$ on the orthonormal latent variable $\mathbf{y}_f = b_f^{\perp}\mathbf{t}_f^{\perp} + \mathbf{f}_f$ equal to $b_f^{\perp} = \mathbf{y}_f^T\mathbf{t}_f^{\perp}$. After F iterations:

$$\mathbf{X} = \mathbf{T}^{\perp}\mathbf{P}^T + \mathbf{E}_F, \mathbf{y} = \mathbf{T}^{\perp}\mathbf{b}^{\perp} + \mathbf{f}_F.$$

At this step conditions (4.10) and (4.12) are satisfied: $(\mathbf{T}^{\perp})^T\mathbf{T}^{\perp} = \mathbf{I}_F$ and $(\mathbf{T}^{\perp})^T\mathbf{f}_F = \mathbf{0}$. Then to provide orthogonality of the matrix $\mathbf{T}^{\perp}$ and the residual matrix $\mathbf{E}_F$ (4.11), let us subtract from the residual matrix its projection on all latent variables $\left\{\mathbf{t}_f^{\perp}\right\}_{f=1}^{F}$: $\tilde{\mathbf{E}}_F = \mathbf{E}_F - \sum_{f=1}^{F}\mathbf{t}_f^{\perp}\left(\mathbf{t}_f^{\perp}\right)^T\mathbf{E}_F$. Thus, $\mathbf{X} = \mathbf{T}^{\perp}\tilde{\mathbf{P}}^T + \tilde{\mathbf{E}}_F$, with a new matrix of loadings $\tilde{\mathbf{P}} = [\tilde{\mathbf{p}}_1, \ldots, \tilde{\mathbf{p}}_F]$, $\tilde{\mathbf{p}}_f = \mathbf{p}_f + \mathbf{E}_F^T\mathbf{t}_f^{\perp}$. The relation $(\mathbf{T}^{\perp})^T\tilde{\mathbf{E}}_F = \mathbf{0}$ holds now. Since conditions (4.10), (4.11) and (4.12) are satisfied, we get $\mathbf{X}^T\mathbf{X} = \tilde{\mathbf{P}}\tilde{\mathbf{P}}^T$ and $\mathbf{X}^T\mathbf{y} = \tilde{\mathbf{P}}\mathbf{b}^{\perp}$. Similar to RPLS for a new data pair $\{\underline{\mathbf{X}}_1, \mathbf{y}_1\}$ regressions on the next data sets will be equivalent:

$$\left\{\begin{bmatrix}\underline{\mathbf{X}}\\ \underline{\mathbf{X}}_1\end{bmatrix}, \begin{bmatrix}\mathbf{y}\\ \mathbf{y}_1\end{bmatrix}\right\} \Leftrightarrow \left\{\begin{bmatrix}\underline{\tilde{\mathbf{P}}}^T\\ \underline{\mathbf{X}}_1\end{bmatrix}, \begin{bmatrix}\mathbf{b}^{\perp}\\ \mathbf{y}_1\end{bmatrix}\right\}.$$

The tensor $\underline{\widetilde{\mathbf{P}}}$ is obtained from the matrix $\widetilde{\mathbf{P}}$, with F as dimensionality of the first mode. Dimensions of the other modes are equal to the dimensions of the other modes of the tensor $\underline{\mathbf{X}}$. Thus, the RNPLS algorithm inherits the tensor representation of data from NPLS and allows effective adaptive learning, which is the property of the recursive PLS.

In addition to identification of the tensor $\underline{\widetilde{\mathbf{P}}}$ and the vector $\mathbf{b}^{\perp}$, which contain information about the training sets on the previous iterations, the algorithm generates also sets of the vectors $\{\mathbf{b}_f\}_{f=1}^{F}$ and the projection vectors $\{\mathbf{w}_f^1, \mathbf{w}_f^2, \mathbf{w}_f^3\}_{f=1}^{F}$, which are used on the prediction stage for determination of an estimation of the depended variable $\hat{\mathbf{y}}$ in the same way as in the traditional NPLS approach (Algorithm 4.3).

The graphical representation of the RNPLS algorithm is shown in Figure 4.13 (see details in Algorithm 4.7).

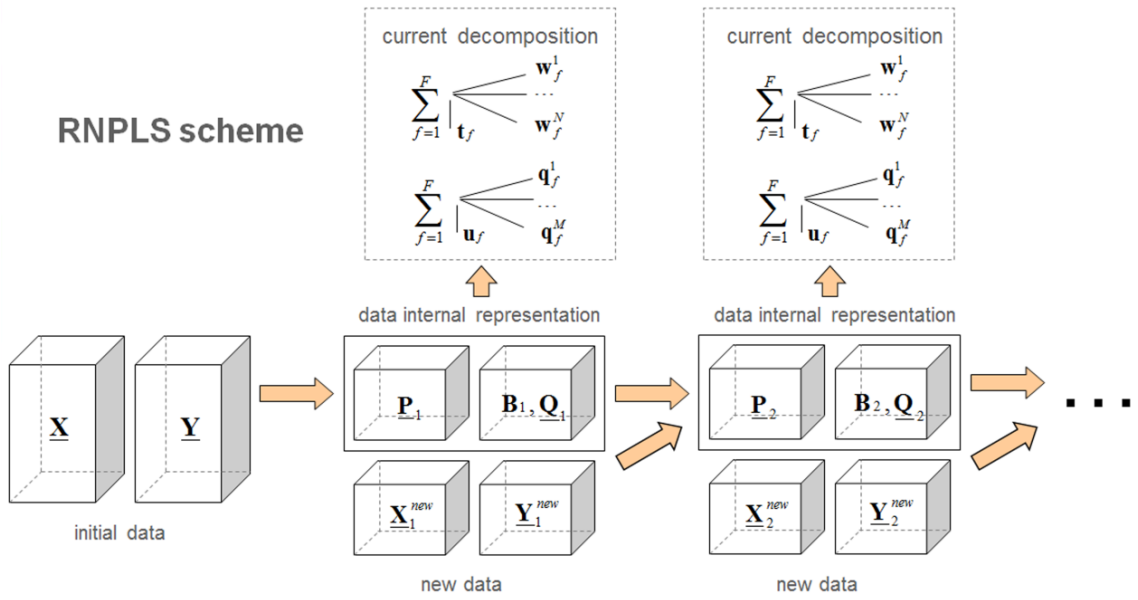

Figure 4.13 The RNPLS scheme. Information used for decomposition of the tensors of observation $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$ is captured by their loading tensors $\underline{\mathbf{P}}$ and $\underline{\mathbf{Q}}$ as well as by the coefficient matrix $\mathbf{B}$ (the general case: $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$ are the tensors of the orders N and M, respectively). In addition, on every iteration the algorithm generates the current sets of the coefficient vectors $\{\mathbf{b}_f\}_{f=1}^{F}$ as well as of the projection vectors $\{\mathbf{w}_f^1, \ldots, \mathbf{w}_f^N\}_{f=1}^{F}$ and $\{\mathbf{q}_f^1, \ldots, \mathbf{q}_f^M\}_{f=1}^{F}$, which are used on the prediction stage for determination of the

estimation of the depended variable $\hat{\mathbf{y}}$ in the same way as in the traditional NPLS approach (see Algorithm 4.3).

Algorithm 4.7 RNPLS

Input: $\underline{\mathbf{X}}^{new}, \mathbf{y}^{new}, \underline{\widetilde{\mathbf{P}}}, \mathbf{b}^{\perp}$; parameters of centering $\underline{\overline{\mathbf{X}}}$, $\bar{y}$, and α; number of factors F.

Outputs: $\{\mathbf{w}_f^1, \mathbf{w}_f^2, \mathbf{w}_f^3\}_{f=1}^{F}$, $\{\mathbf{b}_f\}_{f=1}^{F}$, $\underline{\widetilde{\mathbf{P}}}^{new}$, $(\mathbf{b}^{\perp})^{new}$, $\underline{\overline{\mathbf{X}}}^{new}$, $\bar{y}^{new}$.

1. $\underline{\overline{\mathbf{X}}}^{new} = (1-\alpha)\underline{\overline{\mathbf{X}}} + \alpha \mathrm{Mean}_1(\underline{\mathbf{X}}^{new})$
2. $\forall i: \underline{\mathbf{X}}_{i,:,:,:}^{new} = \underline{\mathbf{X}}_{i,:,:,:}^{new} - \underline{\overline{\mathbf{X}}}^{new}$
3. $\bar{y}^{new} = (1-\alpha)\bar{y} + \alpha \mathrm{Mean}_1(\mathbf{y}^{new})$
4. $\forall i: \mathbf{y}_i^{new} = \mathbf{y}_i^{new} - \bar{y}^{new}$
5. $\underline{\mathbf{X}}_1 = \begin{bmatrix} \underline{\widetilde{\mathbf{P}}}^T \\ \underline{\mathbf{X}}^{new} \end{bmatrix}, \mathbf{y}_1 = \begin{bmatrix} \mathbf{b}^{\perp} \\ \mathbf{y}^{new} \end{bmatrix}$
6. **for** $f = 1$ to F
7. $\quad \underline{\mathbf{Z}}_f = \underline{\mathbf{X}}_f \times_1 \mathbf{y}_f$
8. $\quad \{\mathbf{w}_f^1, \mathbf{w}_f^2, \mathbf{w}_f^3\} = \mathrm{ALS}(\underline{\mathbf{Z}}_f)$
9. $\quad \mathbf{w}_f^m = \frac{\mathbf{w}_f^m}{\|\mathbf{w}_f^m\|}, \quad m = 1,2,3$
10. $\quad \mathbf{X}_f = \underline{\mathbf{X}}_{f\,(1)}$
11. $\quad \mathbf{w}_f = \mathrm{vec}(\mathbf{w}_f^1 \circ \mathbf{w}_f^2 \circ \mathbf{w}_f^3)$
12. $\quad \mathbf{t}_f = \mathbf{X}_f \mathbf{w}_f$
13. $\quad \mathbf{T}_f = [\mathbf{t}_1 | \ldots | \mathbf{t}_f]$
14. $\quad \{\mathbf{T}_f^{\perp}, \mathbf{A}_f\} = \text{QR-decomposition}(\mathbf{T}_f)$: $\mathbf{T}_f^{\perp} = \mathbf{T}_f \mathbf{A}_f$, $(\mathbf{T}_f^{\perp})^T \mathbf{T}_f^{\perp} = \mathbf{I}_f$, $\mathbf{T}_f^{\perp} = [\mathbf{t}_1^{\perp} | \ldots | \mathbf{t}_f^{\perp}]$, $\mathbf{A}_f = [\mathbf{a}_1 | \ldots | \mathbf{a}_f]$
15. $\quad \mathbf{P}_f^T = \mathbf{A}_f^{-1} \mathbf{W}_f^T$
16. $\quad b_f^{\perp} = \mathbf{y}_f^T \mathbf{t}_f^{\perp}$
17. $\quad \mathbf{b}_f^{\perp} = [b_1^{\perp} | \ldots | b_f^{\perp}]^T$
18. $\quad \mathbf{b}_f = \mathbf{a}_f \mathrm{b}_f^{\perp}$
19. $\quad \mathbf{y}_{f+1} = \mathbf{y}_f - b_f^{\perp} \mathbf{t}_f^{\perp}$

20. $\boldsymbol{\tau}_f = \underline{\mathbf{X}}_f \times_4 \mathbf{w}_f^3 \times_3 \mathbf{w}_f^2 \times_2 \mathbf{w}_f^1$

21. $\underline{\mathbf{X}}_{f+1} = \underline{\mathbf{X}}_f - \boldsymbol{\tau}_f \circ \mathbf{w}_f^1 \circ \mathbf{w}_f^2 \circ \mathbf{w}_f^3$

22. **end for**

23. **for** $f = 1$ to F

24. $\mathbf{X}_{F+1} = \underline{\mathbf{X}}_{F+1_{(1)}}$

25. $\widetilde{\mathbf{p}}_f = \mathbf{p}_f + \mathbf{X}_{F+1}^T \mathbf{t}_f^{\perp}$

26. **end for**

27. $\underline{\widetilde{\mathbf{P}}}^{new} = [\widetilde{\mathbf{p}}_1 | \ldots | \widetilde{\mathbf{p}}_F]$, $(\mathbf{b}^{\perp})^{new} = \mathbf{b}_F^{\perp}$.

[a] $\mathrm{Mean}_1(\underline{\mathbf{X}})$ is an average of the tensor $\underline{\mathbf{X}}$ along the first mode.

[b] The n-mode vector product of a tensor $\underline{\mathbf{X}}$ and vector $\mathbf{v} \in \mathbb{R}^{I_n}$ is denoted as $\underline{\mathbf{X}} \times_n \mathbf{v}$ (see Section 0).

[c] $\underline{\mathbf{X}}_{(i)}$ is the mode-i unfolding of the tensor $\underline{\mathbf{X}}$ (see Section 0).

[d] $\mathrm{vec}(\underline{\mathbf{X}})$ is vectorization of the tensor $\underline{\mathbf{X}}$ (see Section 0).

[e] The vector outer product is denoted as $\mathbf{w} \circ \mathbf{v}$ (see Section 4.2.1).

4.9.2 Test of algorithm with simulated data. Comparison with generic NPLS

To study performance and prediction accuracy of RNPLS, we compared it with the traditional NPLS algorithm on simulated data sets with different noise level and for different number of factors. Taking into account specificity of the binary-BCI task ('event'—'non-event' discrimination), the tests were performed only for binary output variables. An artificial data set $\left\{\underline{\mathbf{x}}_k \in \mathbb{R}^{100\times200},\ y_k \in \{0,\ 1\}\right\}_{k=1}^{1600}$ was created in the following way. Binary y_k were randomly generated with equal probabilities. Tensors $\underline{\mathbf{x}}_k$ were calculated according to the rule $\underline{\mathbf{x}}_k = \underline{\mathbf{c}}(y_k) + \lambda\underline{\varepsilon}_k$, where $\underline{\mathbf{c}}(y_k) \in \mathbb{R}^{100\times200}$ is set as $c_{ij} = \cos(2.5\pi(i - 0.4j) + 2)$ if $y_k = 0$, and $c_{ij} = \sin(2.5\pi(i + 0.4j) + 1)$ if $y_k = 1$ (Figure 4.14). The random noise $\underline{\varepsilon}_k \in \mathbb{R}^{100\times200}$ was drawn from a multivariate normal distribution $\mathcal{N}(\underline{\mathbf{0}}, \underline{\boldsymbol{\Sigma}})$. It was added to the templates with a parameter λ introducing the signal-to-noise ratio: $\lambda = \{0.5;\ 1;\ 5;\ 10\}$. The noise has the same

amplitude as the signal $\underline{\mathbf{c}}(y_k)$ in the case when $\lambda = 1$. The entire data set was split into the training and the test data sets of equal size.

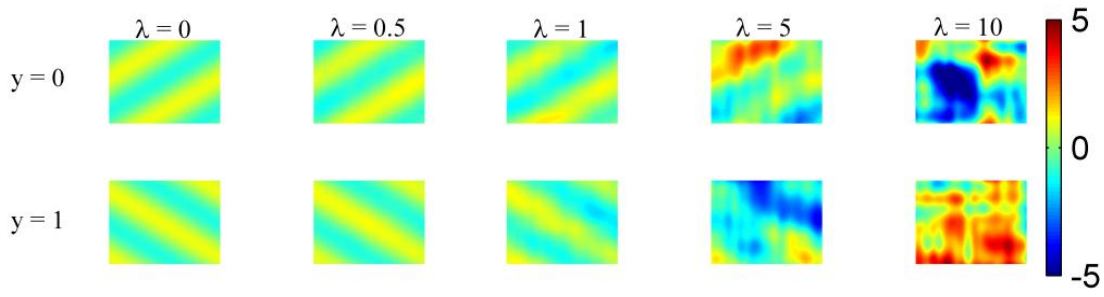

Figure 4.14 Example of points $\{\underline{\mathbf{x}}_k, y_k\}$ from the artificial dataset, with different levels of noise.

The NPLS algorithm processed the whole training set. For the recursive calculation with RNPLS, the training set was split into 20 disjoint windows, each one containing 40 points. For all conditions (the noise level and the number of factors), the experiment was repeated 10 times with new realizations of noise. Then the predictions of the output variable were averaged over these 10 experiments. The resulting percentage of prediction errors is shown in (Figure 4.15). In all the tests, RNPLS demonstrated significantly smaller number of factors, which are necessary for efficient prediction. Moreover, it showed better robustness. Variation of the prediction errors was essentially smaller for the recursive algorithm for small number of factors for noise level up to 500%. Advantages of RNPLS can be explained by overfitting effect suppression.

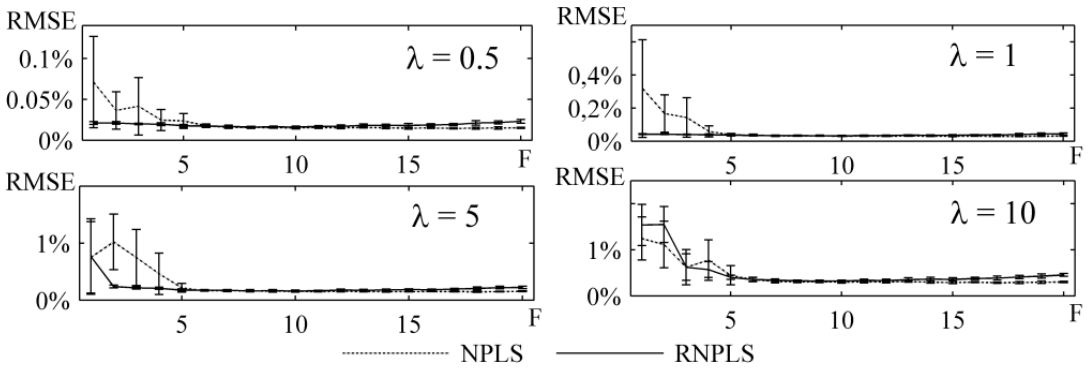

Figure 4.15 Comparison of prediction errors (root mean squared error, RMSE) for the NPLS and the RNPLS algorithms on the test sets for different levels of noise λ and different number of used factors *F*.

4.9.3 Convergence of RNPLS

To study convergence properties of the RNPLS method, a set of computational experiments were carried out with artificial data described in Section 4.9.2.

At each iteration the RNPLS algorithm generates the current sets of coefficients $\{\mathbf{b}_f\}_{f=1}^{F}$ as well as projection vectors $\{\mathbf{w}_f^1, \dots, \mathbf{w}_f^{\mathrm{N}}\}_{f=1}^{F}$ and $\{\mathbf{q}_f^1, \dots, \mathbf{q}_f^{\mathrm{M}}\}_{f=1}^{F}$ (see Figure 4.13), which can be used to predict of an estimation $\underline{\hat{\mathbf{y}}}(t)$ of the dependent variable $\underline{\mathbf{y}}(t)$ from the independent variable $\underline{\mathbf{x}}(t)$. The model can be represented as linear in the original variables $\underline{\mathbf{x}}(t)$ and $\underline{\mathbf{y}}(t)$. For the case of scalar $y(t) \in \mathbb{R}$ and matrix $\mathbf{x}(t) \in \mathbb{R}^{I_1 \times I_2}$ (see the dataset from Section 4.9.2) linear relations between them can be represented as:

$$\hat{y}(t) = \sum_{i,j} x_{i,j}(t) c_{i,j}, \mathbf{C} = \{c_{i,j}\}_{i,j}.$$

In our computational experiments $\mathbf{x} \in \mathbb{R}^{100\times200}$, $y \in \{0,1\}$, and the regression coefficient matrix $\mathbf{C} \in \mathbb{R}^{100\times200}$. To analyze the convergence of the set of matrixes $\{\mathbf{C}_i\}$ we introduce the distance between two matrices $\mathbf{C}_i$ and $\mathbf{C}_j$ as:

$$\rho(\mathbf{C}_i, \mathbf{C}_j) = \frac{\|\mathbf{C}_i - \mathbf{C}_j\|_{\mathrm{F}}}{I_1 \cdot I_2},$$

where, $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm (see Equation (4.4)).

To study the convergence of the RNPLS algorithm computational experiments were carried out. A training set consisting of 800 points was split on 40 disjoint subsets

(each one contains 20 points). The level of noise was taken equal to 1000% of the signal amplitude ($\lambda = 10$). The number of factors $F = 9$ was chosen in a way to minimize the average RMSE on the test set over 10 realizations of the training set (see Figure 4.15). Convergence of the matrixes is represented in Figure 4.16. As it could be seen from Figure 4.17, the changes in the regression coefficients matrix $\mathbf{C}$ become insignificant after the 10-th iteration.

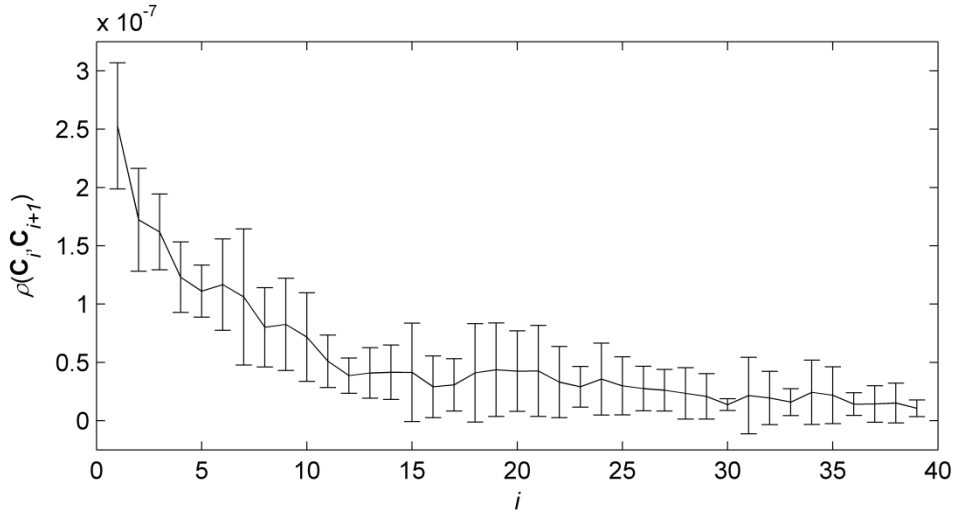

Figure 4.16 Convergence of the coefficient matrices versus the iterations of the RNPLS algorithm. The distance ρ between two successive coefficient matrices decreases significantly after the first 10 iterations.

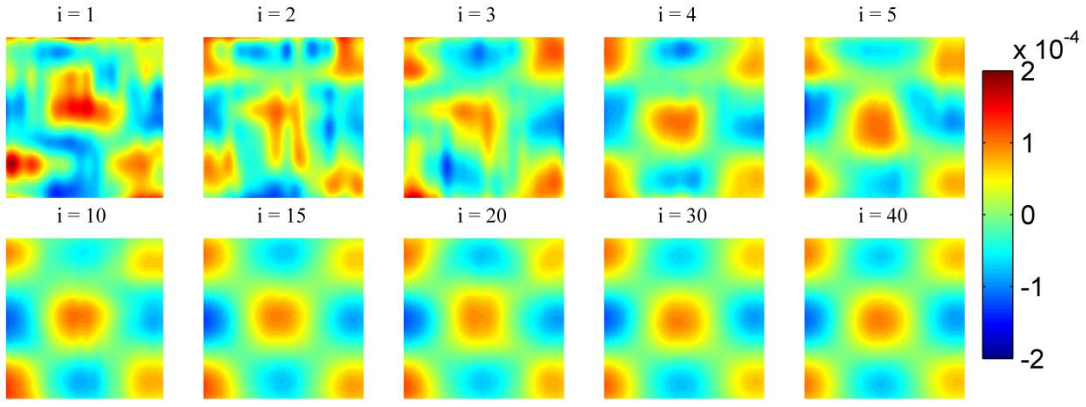

Figure 4.17 The regression coefficients $\mathbf{C}$ depending on the iteration number i of the RNPLS algorithm.

To compare RNPLS and NPLS, both algorithms were applied to 10 realizations of the training set (the level of the noise $\lambda = 10$). The factor numbers were chosen equal to 9 and 17 for RNPLS and NPLS, respectively, in a way to minimize the average RMSE on the test set over 10 realizations of the training set (see Figure 4.15). Images of the obtained regression coefficients $\{\mathbf{C}_i^{RNPLS}\}_{i=1}^{10}$ and $\{\mathbf{C}_i^{NPLS}\}_{i=1}^{10}$ are represented in Figure 4.18. Variability of the NPLS results ($\sigma_{NPLS} = 1.59 \cdot 10^{-5}$) is 2.4 times greater than variability of the RNPLS results ($\sigma_{RNPLS} = 6.6 \cdot 10^{-6}$). It can be explained by the fact that NPLS is more affected by noise as well as overfitting effect.

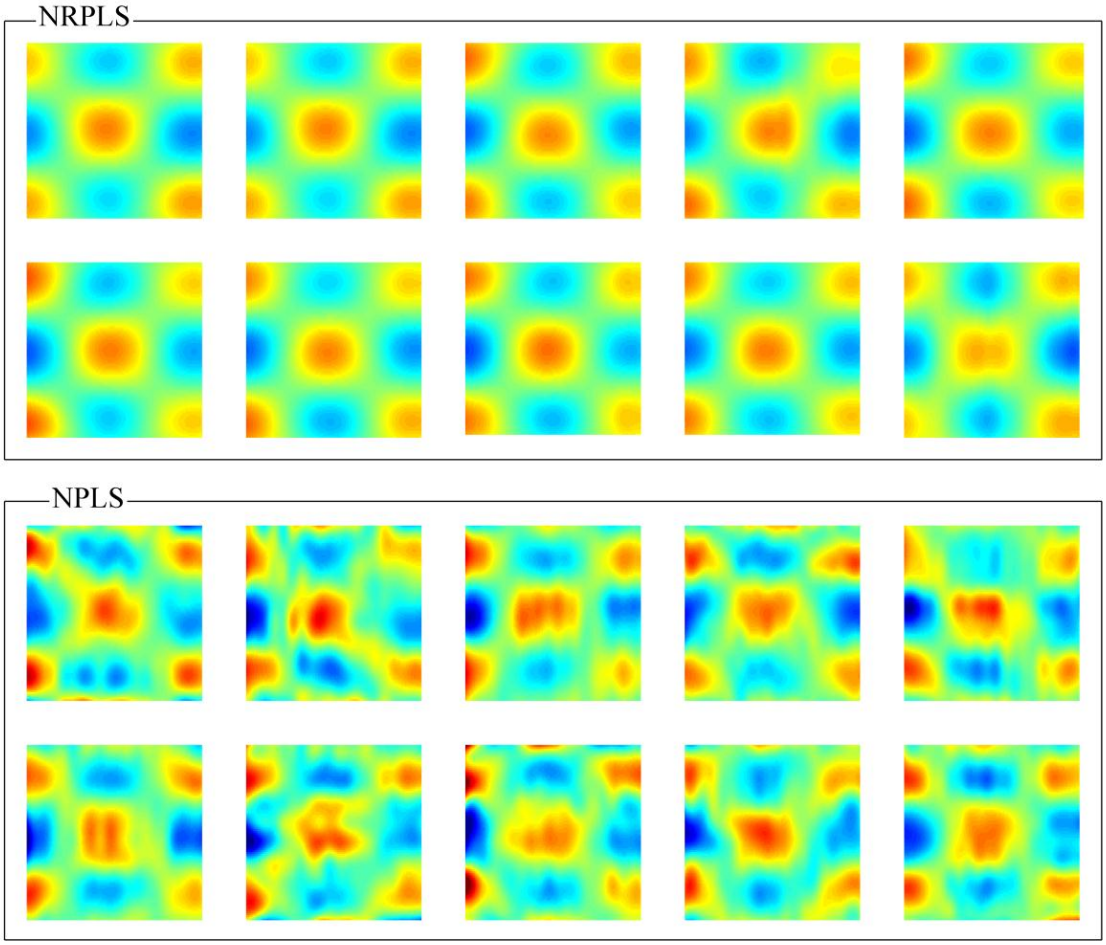

Figure 4.18 The regression coefficients defined by the RNPLS and NPLS algorithms in the 10 realization of the training set ($\lambda = 10$).

For comparison, the ordinary PLS coefficients were computed for the training set without noise ($\lambda = 0$). Figure 4.19 represents the regression coefficients for the RNPLS and the NPLS methods (average over 10 realizations, $\lambda = 10$), they are compared to the

PLS coefficients. It could be seen by visual expectation that the result generated by RNPLS is more close to the PLS one than the NPLS solution.

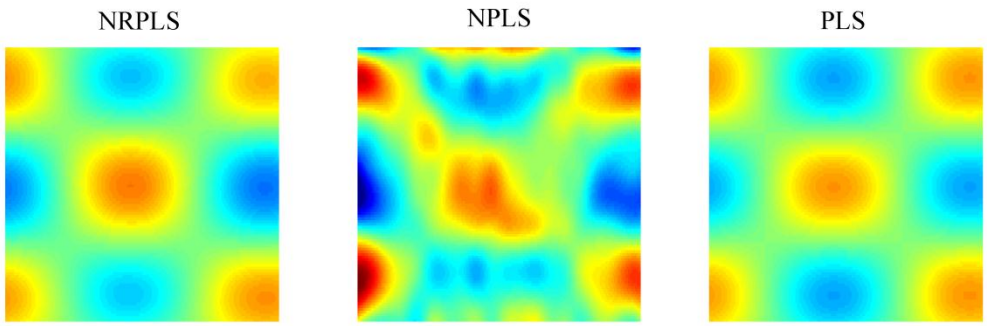

Figure 4.19 Comparison of the regression coefficients defined by the RNPLS and the NPLS algorithms (average over 10 realizations, $\lambda = 10$) with the ones generated by PLS for undisturbed classes (noise is absent).

The next step is the comparison of the estimations found by the NPLS and the RNPLS approaches with the one obtained by the PLS methods. Behavior of the mean values and the standard deviations of the $\rho\left(\mathbf{C}_{\lambda=0}^{PLS}, \mathbf{C}_{\lambda>0}^{RNPLS}\right)$ for the different noise levels λ over the algorithms iterations are shown in Figure 4.20 (the statistics calculated for 10 data realizations). In parallel, the means and the standard deviations of $\rho\left(\mathbf{C}_{\lambda=0}^{PLS}, \mathbf{C}_{\lambda>0}^{NPLS}\right)$ computed for the data generated by the NPLS algorithm is represented for every level of noise. As it could be seen, the mean value of the distances between the RNPLS and PLS regression coefficients already after 1-2 iterations (20-40 points) are significantly less than the ones between the NPLS and PLS regression coefficients computed for the whole training sets (800 points). In addition, the standard deviations of the RNPLS results are generally less than those obtained by NPLS.

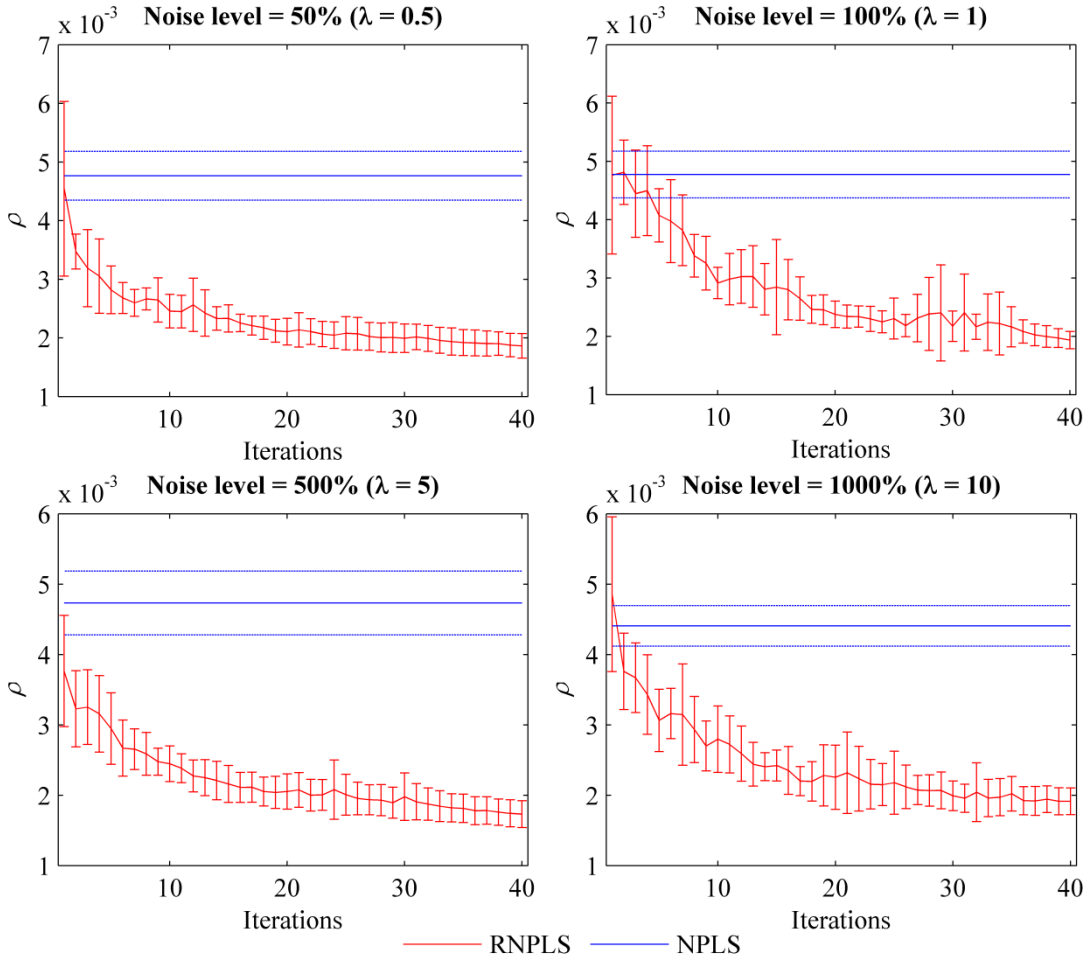

Figure 4.20 Red lines: the mean values and the standard deviations (for 10 experiments) of the distances between the regression coefficients generated by the RNPLS and the ordinary PLS $\rho(\mathbf{C}^{PLS}, \mathbf{C}^{RNPLS})$ for different noise levels over the RNPLS algorithms recursive iterations. Blue lines: the mean values and the standard deviations (for 10 experiments) of the distances between the regression coefficients generated by the NPLS (applied for whole training sets) and the ordinary PLS $\rho(\mathbf{C}^{PLS}, \mathbf{C}^{NPLS})$ for different levels of noise.

In conclusion, as it was shown in the set of computational experiments, both the RNPLS and the NPLS algorithms approximate the PLS solution obtained in the case when the noise was absent. RNPLS approach demonstrates faster convergence (20-40 points instead of 800 for NPLS) and is considerably noise-steady. For all tested levels of noise the disturbance in the RNPLS solution was appreciably less relatively to NPLS.

4.10 Conclusions

Multi-way approach is promising for BCI tasks. It allows simultaneous analysis of different modalities, preserves the structure of the data, improves robustness of results, as well as allows identifying relative impact of each feature on the final model. PARAFAC and NPLS are widespread project methods for the case of highly correlated multi-way (tensor) data of high dimensionality. However, NPLS constructs the projectors by taking into account information about the response variables. Thus, it was chosen as the basis approach in this study. The shortcomings of the NPLS algorithm are excessive complexity of the identified model, method's computational efficiency and resources consumption. A set of modifications of NPLS were proposed in this chapter: L1-Penalized NPLS (providing an opportunity for selection of informative features), Iterative NPLS (allows storing only part of the processing data in the active memory) and Recursive NPLS (provides an opportunity for online treatment of the data). Suggested algorithms were tested in the computational experiments on artificial datasets. Experiments demonstrated promising results. Thus, the methods could be efficiently applied for the BCI tasks. However, the proposed approaches have some drawbacks. For instance, the INPLS algorithm cannot treat data flow, L1-Penalized NPLS gives a sparse solution that could be not appropriate for some modalities (e.g., frequency), and RNPLS could bring to some deterioration of the prediction quality.

Chapter 5. APPLICATION OF MULTIMODAL ANALYSIS TO BCI EXPERIMENTS

BCI experiments (see Chapter 2), based on a simple reward oriented task, were carrying out by the Clinatec neurological team. Each animal has the opportunity to push a wall-mounted pedal. Every pushing event activates the food dispenser and the animal obtains a reward (food pellet or juice). The animal was trained to press the pedal without any cue or conditioning stimulus.

According to Chapter 2, BCI experiment consists of two stages. The first one is the Training Stage. During this stage, the training dataset is collected for the following BCI system calibration. Namely, the ECoG signals and the pedal signal are recorded simultaneously during the behavior experiments described in Chapter 2. During the first stage, the food dispenser is activated by the pedal. Those training sets which contain less than 50 pushes are neglected. The offline calibration procedure based on the collected recordings results in the event predictor computation.

During the second stage, the Close Loop Real-Time BCI, the predictor is applied immediately to the signal from the animal's brain in real-time. The animal presses the pedal but it obtains the reward only in the case when the predictor detects the pushing event. More precise description of the experimental setup is given in Chapter 2.

5.1 Signal acquisition and pre-processing

A set of electrodes was implanted on the surface of the rat's and the monkey's cortex for the neural activity recording. The ECoG signals were registered at 6.5 kHz, or 13 kHz sampling rate either with the Biomea® system (Biologic, Grenoble, France), or at 1 kHz by the Micromed® system (Micromed SD64, Micromed Italy). In parallel, the SD32 system (Micromed, Italy) records and displays the signals. During the experiments the rat was freely moving in the Abet II behavioral cage whereas the

monkey was sitting in a custom made primate chair minimally restrained and its neck was collar hooked to the chair.

The signal was downsampled to 1.3 kHz or 1 kHz. The Common Average Reference (CAR) filter was applied to the signal of the rat, i.e., average signal among all the electrodes was subtracted to eliminate a 'common source' (Ludwig et al., 2009). Application of this filter is made due to the presence of the strong noise signal sources whereas the signal of interest is relatively weak in the case of the rat.

5.2 Feature extraction

Continuous Wavelet Transform (CWT) was applied for feature extraction:

$$x(t) \Leftrightarrow W_\psi\{x\}(a,\tau),$$
$$W_\psi\{x\}(a,\tau) = \left(x, \psi_{a,\tau}\right) = |a|^{\frac{1}{2}} \int_{-\infty}^{+\infty} x(t)\overline{\psi\left(\frac{t-\tau}{a}\right)}\,dt, \tag{5.1}$$
$$\psi_{a,\tau}(t) = \psi\left(\frac{t-\tau}{a}\right)$$

where a and τ represent the scale and the shift in the time-frequency domain, respectively. $\psi(t)$ is a wavelet function, which effectively limits duration and has an average value of zero. For more information, see Teolis (1998).

In the present study, Meyer wavelet (Figure 5.1) was chosen as a mother wavelet taking into account its computational efficiency (Sherwood and Derakhshani, 2009). Analytical representation of the wavelet in the frequency domain is given by equation (5.2).

$$\hat{\psi}(\omega) = \begin{cases} |2\pi|^{-1/2} e^{i\omega/2} \sin\left(\frac{\pi}{2}\nu\left(\frac{3}{2\pi}|\omega| - 1\right)\right), & if \quad \frac{2\pi}{3} \le |\omega| \le \frac{4\pi}{3} \\ |2\pi|^{-1/2} e^{i\omega/2} \sin\left(\frac{\pi}{2}\nu\left(\frac{3}{4\pi}|\omega| - 1\right)\right), & if \quad \frac{4\pi}{3} \le |\omega| \le \frac{8\pi}{3}, \\ 0, & if \quad |\omega| \notin \left[\frac{2\pi}{3}, \frac{8\pi}{3}\right] \end{cases} \tag{5.2}$$
$$\nu(c) = c^4(35 - 84c + 70c^2 - 20c^3),\ c \in [0,\ 1]$$

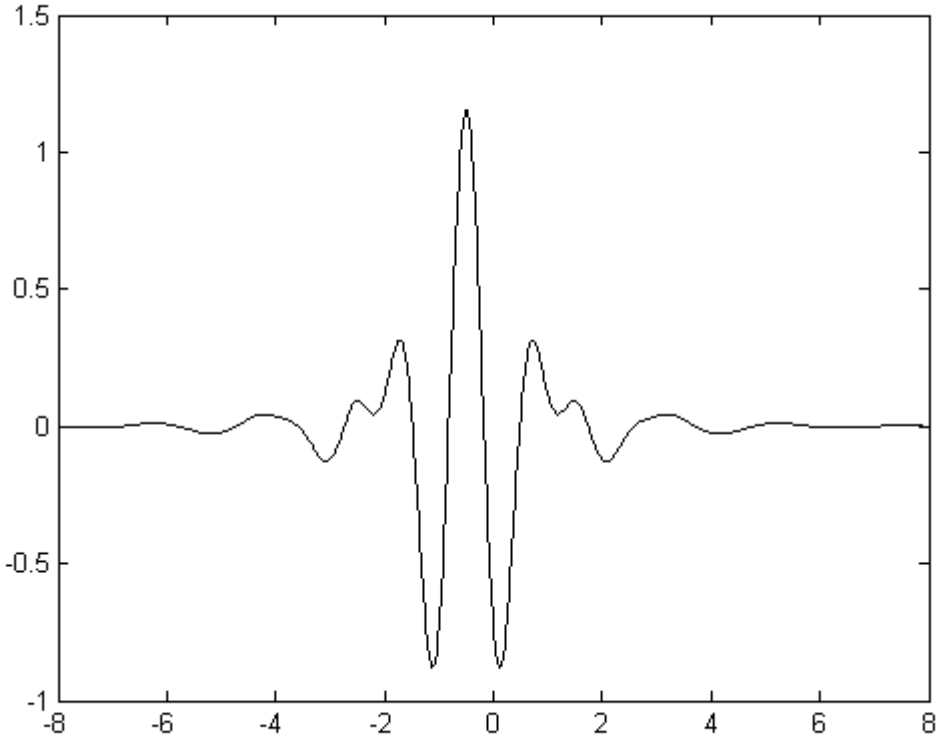

Figure 5.1 Meyer wavelet.

5.3 Tensor representation of the ECoG data

To form the BCI feature tensor, each ECoG epoch sample was mapped to the temporal-frequency-spatial space by CWT. For each epoch j (determined by its final moment t), electrode c, frequency f and time shift τ, elements $x_{j,\tau,f,c}$ of the tensor $\underline{\mathbf{X}}$ were calculated as the norm of CWT coefficients of the ECoG signal (see Figure 5.2). Frequency band $[10,\ 300]$ Hz with step $\delta f = 2$ Hz was chosen for both monkey and rat. The analyzed time interval was equal to $\Delta\tau = 2$ s (with resolution $\delta\tau = 0.04$ s) for the rat and $\Delta\tau = 0.5$ s (with resolution $\delta\tau = 0.01$ s) for the monkey (according to the biological point of view). For the case of the rat $c = \overline{1{,}14}$, and $c = \overline{1{,}32}$ for the case of the monkey. The resulting dimensions are $(146 \times 51 \times 14)$ and $(146 \times 51 \times 32)$, respectively. The binary dependent variable was set to one, $y_j = 1$, if the pedal was pressed at the moment t, and $y_j = 0$, otherwise.

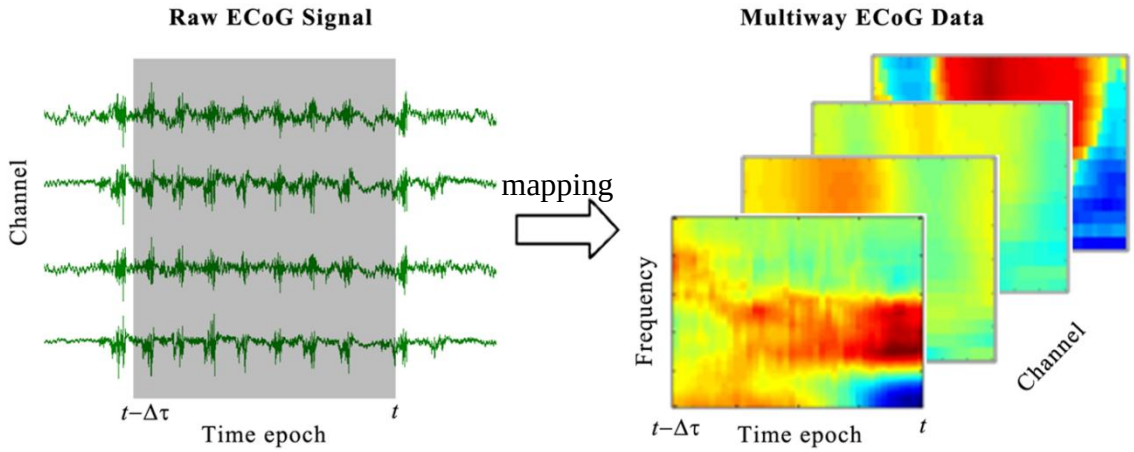

Figure 5.2 Time epoch of the multi-channel ECoG recording mapped by continuous wavelet transform to the temporal-frequency-spatial feature space.

5.4 BCI system calibration

The tensor $\underline{\mathbf{X}}$ and the vector **y**, representing the training dataset, were used for the BCI system calibration. Because of huge dimension of the feature tensor $\underline{\mathbf{X}}$, an iterative method was employed. In order to apply the INPLS algorithm, the training dataset was split into subsets in such a way that each one consisted of 30 points. Total number of the points in the training dataset was $n = 1500$, including all event-related points and randomly selected non-event ones. The number of factors was chosen by ten-fold cross validation ($F = 5$ for the rat and $F = 7$ for the monkey). On the basis of the given data INPLS identified a predictive regression model between the dependent variable y and the tensor $\underline{\mathbf{x}}$ of independent variables. In addition a set of projection vectors for each modality was generated. Let us note that each factor could be considered as a spatial-frequency-temporal filter, formed as outer product (Kolda and Bader, 2007) of particular filters related to each modality: spatial, frequency, and temporal.

The overall goal of our study was to calibrate the BCI system using the signal of a single electrode. For this purpose, in the case of experiments in rats, the most informative electrode was chosen according to the Modality Influence Analysis, as the one with the highest weight across all the electrodes (see Section 4.6.3). In the case of monkey, the Recursive NPLS algorithm was applied to select the most informative electrode, since RNPLS has demonstrated better efficiency in comparison with MI

analysis when number of electrodes was great. The training tensors related to the selected electrodes were analyzed with INPLS to construct the single-electrode predictors.

Let us note that we estimate the dependent variable $\hat{y}(t)$ as a linear combination of the components of the ECoG signal decomposition at the time moment t: $\hat{y}(t) = \sum_f \sum_\tau b_{f,\tau} x_f(t-\tau) + b_0$. Here, $x_f(t)$ is the absolute value of the continuous wavelet decomposition coefficient for the frequency f at the moment t. Thus, the resulting model presets a dynamical filter.

For binarization of the predicted output variable $\hat{y}(t)$, a scalar threshold was chosen maximizing the overall performance ($OP = (TPR + PPV)/2$, see Section 1.2.5) of the BCI algorithm on the training data set running in the online mode simulation.

Calibration was made on one recording. The resulting model was tested in all recorded datasets as well as in the online experiments.

5.5 Simulation of BCI experiments

Offline emulations of BCI experiment, were fully reproducing online sessions and were carried out to study a generalization ability of the predictive model. For this purpose, the set of recordings of behavioral experiments were played back. The decisions (‘event’ or ‘non-event’) were made sequentially for each half-a-second-length buffer of the recordings. This corresponds to the real-time data acquisition in the CLINATEC/LETI/CEA BCI experimental platform. The predictors were calculated every 0.125 s, i.e., 4 times per buffer. The buffer was considered as containing the ‘event’ if at least one of these predictors surpassed the threshold of binarization. After each detection, the system was blocked for 5 seconds in the case of the rat and 1.5 seconds in the case of the monkey to prevent multiple activations. Following to (Fatourechi et al., 2008) the real event was considered as detected (True Positive, see Section 1.2.5) if the time interval between the real event and its detection did not exceed 1.5 s (Figure 1.19).

5.6 Proof and comparison of solutions

To confirm the selected approaches and implementations, a set of comparative experiments were carried out.

The present section consists of three parts. In the first one, several mother wavelets were compared by means of correlation between the wavelet coefficients and the signal of the pedal. For this purpose, we used the series of recordings, made during the long-term experiments in the rat freely moving in the natural-like environment. The second part is a comparison of the unsupervised (PARAFAC) and supervised (INPLS) approaches for classification of the real BCI data recorded in the experiment in monkey. The last part is devoted to the study of the efficiency of different classifiers in the space of the INPLS latent variables. Simulations of the online experiments were made on the basis of 9 recordings of the experiments in the rat.

5.6.1 Comparison of different mother wavelets

To project recorded data to the feature space continuous wavelet transform is applied. To found the most effective function for this projection several mother wavelets ψ were compared, namely, Meyer, Morlet, Symlet '7' and '8', 2nd and 10th orders Debauchies, Coiflets '5', and Haar (Figure 5.3).

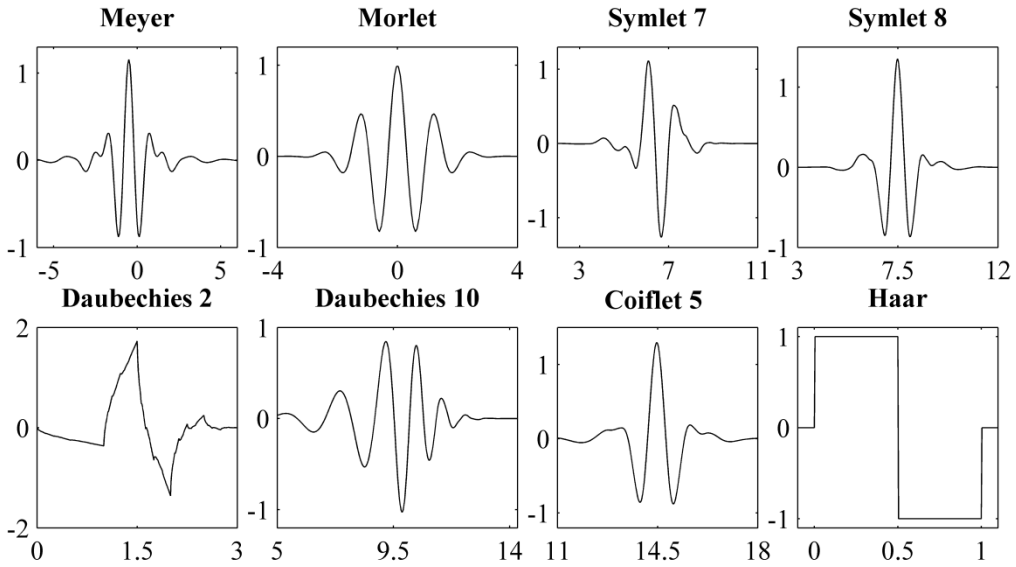

Figure 5.3 ψ-functions of the examined mother wavelets.

Evaluation of the wavelets was made in a set of simulations of the BCI experiments. Altogether, 4 different recordings of the rat's brain neural activity received 9/7/2009 and 16/7/2009 were used for the test. Comparison was made according to the maximum level of correlation between the absolute value of the wavelet's coefficients

$W_\psi(a,\tau)$ (taking over the scale factors a and the time shifts τ, see Section 5.2), and the signal of the pedal y:

$$R_\psi = \max_{a,\tau}\left\{\underset{t}{\text{corr}}\left(\left|W_\psi(a,t-\tau)\right|, y(t)\right)\right\},$$

$$\underset{t}{\text{corr}}\big(u(t),v(t)\big) = \frac{\sum_{i=1}^{n}(u(t_i)-\bar{u})(v(t_i)-\bar{v})}{\sqrt{\sum_{i=1}^{n}(u(t_i)-\bar{u})^2\sum_{i=1}^{n}(v(t_i)-\bar{v})^2}},$$

$$\bar{u} = \frac{1}{n}\sum_{i=1}^{n}u(t_i),\ \bar{v} = \frac{1}{n}\sum_{i=1}^{n}v(t_i),$$

where a corresponds to the frequencies of the band [10,300] Hz and $y(t) = \{0,1\}$ represents the position of the pedal at the moment t.

Comparison of the mother wavelets shows that 2nd order Daubechies and Haar lead to a relatively low level of correlation, whereas the performance of all other wavelets is comparable. Meyer wavelet was chosen for the present study as the mother function, due to its computational efficiency (Sherwood and Derakhshani, 2009). Results are shown in Figure 5.4.

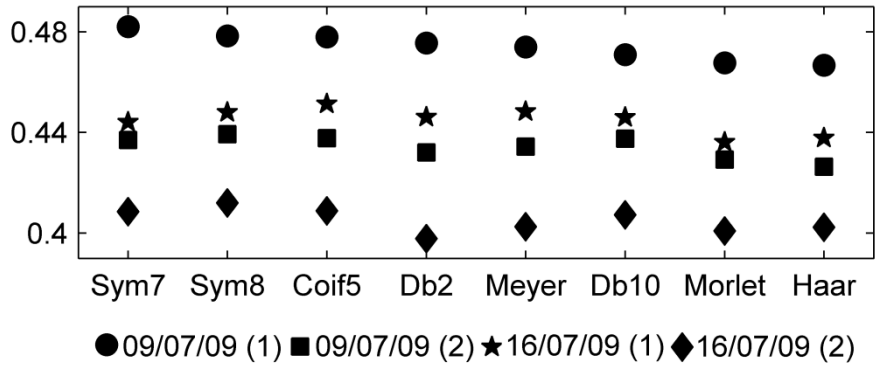

Figure 5.4 Maximum of correlation between the wavelet coefficients and the signal of the pedal for different types of the mother wavelets.

5.6.2 Comparison of unsupervised and supervised classifiers

Unsupervised methods, like PARAFAC, Tucker Decomposition, or Non-negative Tensor Factorization are often used to form a feature space in the tasks with multimodal data (Nazarpour et al., 2006; Zhao et al., 2009; Mørup et al., 2008). These approaches could be effectively applied for dimensionality reduction by means of selection of a set of directions with largest variation of the data. At the same time, these

methods do not take into account distribution of the classes along the considered directions. Thus founded projectors could be not effective enough for the purposes of separation of different classes. For this reason, supervised methods of the feature space formation (e.g., General Tensor Discriminant Analysis (Li et al., 2009), Regularized Tensor Discriminant Analysis (Li and Zhang, 2010), N-way PLS (Bro, 1996)) are preferable for the BCI tasks. In this section we compare effectiveness of PARAFAC combined with a linear regression and the NPLS algorithm on the real data recorded in the experiment in monkey. The main difference in the applied approaches is that whereas the NPLS algorithm simultaneously finds the space of latent variables and builds a linear regression, PARAFAC independently projects data into the space of latent variables (identified without any information about the classes) and after we perform a linear multiple regression on the obtained projections.

During it the monkey was pressing the pedals which were located in different positions. The recorded data for two positions of the pedal (‘left’ and ‘up’) were split on the training and the test datasets of the same size. Comparison of the methods consists of two parts. The first one is separation of the background monkey’s activity from the pressings of the ‘left’ pedal. The second one is discrimination of the ‘left’ pressings from the ‘up’ ones. The number of factors was ranging from 1 to 20.

Figure 5.5 demonstrates the root mean squared error (RMSE) calculated on the test set by both methods for separation of the ‘non-event’ and the ‘left-event’ epochs. The training and the test sets contain 350 points (50 ‘events’ and 300 ‘non-events’. The optimum achieved by the NPLS approach outperforms the one of PARAFAC in terms of the RMSE, at the same time NPLS requires less number of factors to achieve the extremum.

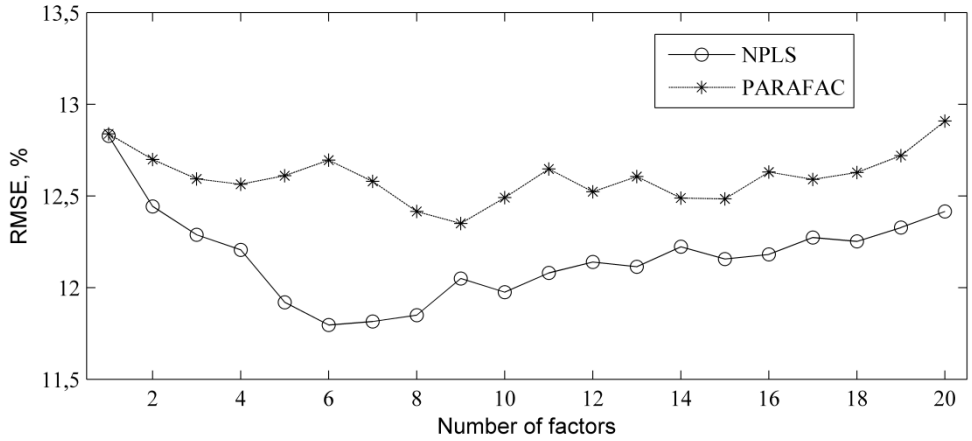

Figure 5.5 Comparison of RMSE values for classification ‘non-event’ vs. ‘left-event’ performed by the NPLS and the PARAFAC approaches for different numbers of factors. NPLS outperforms PARAFAC in terms of RMSE, moreover it requires less number of factors to achieve the optimal classification results.

Figure 5.6 represents the RMSE calculated on the test for discrimination of the ‘up-events’ from the ‘left-events’. The training consists of 100 points (50 ‘left-events’ and 50 ‘up-events’), the test set holds 100 points (50 ‘left-events’ and 50 ‘up-events’). This experiment also has demonstrated better efficiency of the NPLS algorithm for classification tasks in comparison with PARAFAC approach.

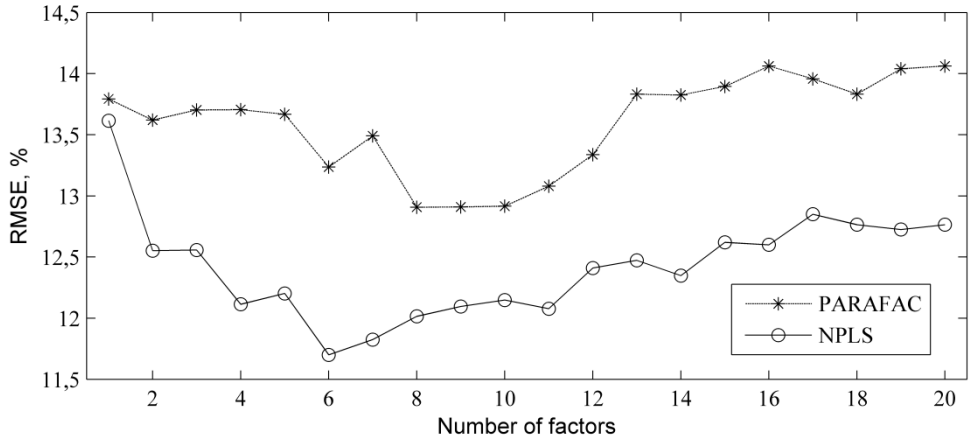

Figure 5.6 Comparison of RMSE values for classification ‘up-event’ vs. ‘left-event’ performed by the NPLS and the PARAFAC approaches for different number of factors.

5.6.3 Comparison of different classifiers

The NPLS algorithm is based on the data projection to the low dimensional feature space (the space of latent variables), with simultaneous construction of a linear regression. The binary BCI leads to the problem of two-classes discrimination. Although the PLS algorithm (as well as NPLS and its derivations) was not inherently designed for classification, it is widely applied to solve this problem (Barker and Rayens, 2003). In particular in the studies (Eliseyev et al., 2011a; Chao et al., 2010; Acar et al., 2010) the NPLS linear regression was applied for classification using binary output variables. Otherwise PLS was used as a dimensionality reduction tool and coupled with different classifiers in the space of latent variables (for more details see Barker and Rayens, 2003). The goal of the present section is to study the efficiency of different classifiers in the space of the INPLS latent variables in the context of the binary self-paced BCI. For the comparison of classifiers, we used the recordings from the series of 9 long-term experiments in the rat freely moving in the nature-like environment.

Different types of classification methods have been applied in the BCI tasks (Lotte et al., 2007). Several linear and non-linear classifiers widely used in the BCI research were chosen in this study. They were compared using a given set of the INPLS features.

Linear classifiers:

- *Linear Discriminant Analysis (Fisher's LDA):* in the space of the latent variables it constructs a hyperplane to separate two classes (Duda et al., 2001; Fukunaga, 1990). The method assumes normal distribution of the data, as well as equal covariance matrix for both classes. The separation hyperplane is seeking to maximize the distance between the classes and minimize the interclass variance.
- *INPLS regression with binarization:* the INPLS algorithm generates a linear regression model in the latent variables space to predict $\hat{\mathbf{y}}$ corresponding to the output variable $\mathbf{y}$. For binarization of $\hat{\mathbf{y}}$ a scalar threshold was found using the training set by means of maximization of the Overall Performance (OP).

Non-linear classifiers:

- *Quadratic Discriminant Analysis:* the method assumes different normal distributions of data in the different classes (Duda et al., 2001; Fukunaga, 1990). The

Bayes rule is used to compute a posteriori probability that point belongs to a given class. The task leads to a quadratic decision surface, which explains the name of the classifier.

- *Logistic Regression:* the method assumes binomial distribution of the data (Hilbe, 2009). The model supposes that probability that the point belongs to the first class can be expressed as $\Pr(y = 1|\mathbf{x}) = 1/\big(1 + \exp(-z)\big)$, where z is a linear function of $\mathbf{x}$. The unknown parameters of this linear function are usually estimated on the training set by maximum likelihood procedure.
- *Kernel Support Vector Machine:* the method uses a discriminant hyperplane to identify the classes (Cortes and Vapnik, 1995), which is found in the way to maximize the margin, i.e., the distance between the nearest training points belonging to the different classes. By using a 'kernel trick' (Aizerman et al., 1964) the method could be applied as non-linear classifier, by mapping of the data points to higher dimension space. In the experiment we have chosen the Gaussian kernel (radial basis function, RBF) with the parameter $\sigma = 1$.

Training of all classifiers was carried out using the training dataset. Then the efficiency of the classifiers in the BCI task was estimated by simulation of the BCI experiments using the test recordings. Binary discriminators were applied offline to 9 recordings (lasting from 10 minutes to 1 hour).

Table 5.1 summarizes the results of comparison of classifiers in the simulation of the online BCI experiments. The simulations show that the quadratic classifier (QDA) applied to the INPLS latent variables is the most efficient. Nevertheless, this method does not significantly outperform the linear regression with binarization threshold used by INPLS. Thus, using of the INPLS algorithm without application of any additional classifier is reasonable. The values of the Overall Performance (Section 1.2.5) for 9 test datasets and all classifiers are shown in Figure 5.7.

Table 5.1 Performance of the simulated self-paced BCI experiments with different classifiers in the space of the latent variables.

	TPR (%)	PPV (%)	FPR (%)	ERR (%)	OP (%)	FP min^{-1}
INPLS	80 ± 14	87 ± 9	0,36 ± 0,16	0,91 ± 0,33	84 ± 11	0,42 ± 0,19
LDA	77 ± 13	81 ± 13	0,55 ± 0,21	1,21 ± 0,37	79 ± 13	0,64 ± 0,25
QDA	**85 ± 13**	**86 ± 9**	**0,45 ± 0,29**	**0,85 ± 0,31**	**85 ± 10**	**0,52 ± 0,34**
KSVM	76 ± 12	81 ± 13	0,57 ± 0,43	1,31 ± 0,43	79 ± 11	0,67 ± 0,50
Logit	83 ± 14	83 ± 11	0,52 ± 0,27	0,97 ± 0,31	83 ± 12	0,61 ± 0,31

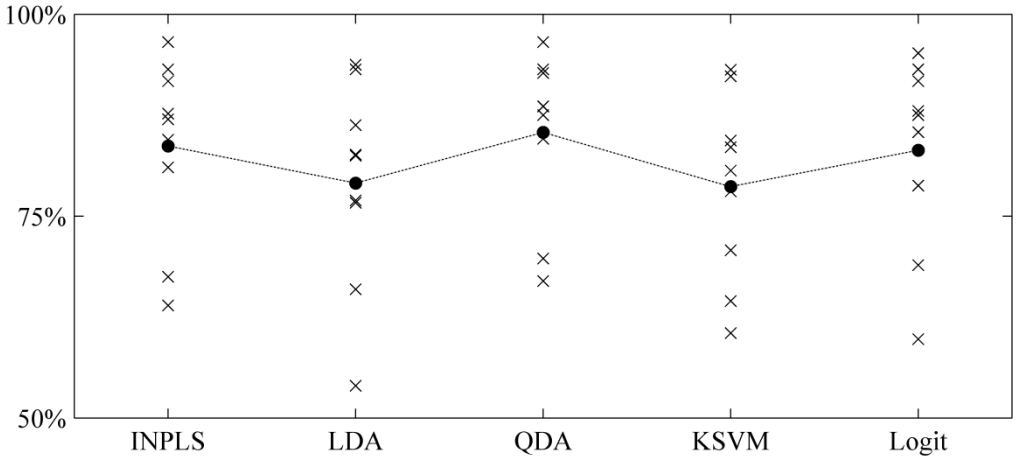

Figure 5.7 The Overall Performance (OP) for the series of the simulated self-paced BCI experiments using different classifiers in the space of the INPLS latent variables. Black circles represent the average value over all the experiments.

5.7 Conclusion

In this chapter, structural parameters of the applied method were compared in the simulation of the online BCI experiments.

Comparison of different mother wavelets used for mapping of the ECoG signals has demonstrated similar performance of all bases of decomposition. Thus, taking into account its computational efficiency, Meyer wavelet was chosen.

A set of test experiments has confirmed that supervised classifier (INPLS), which took into account information about classes, was preferable than unsupervised classifier (based on PARAFAC). Thus, the initial choice of the supervised method was justified.

Different classifiers in the space of the latent variables were compared in a set of simulations of the online BCI experiments. The quadratic classifier has demonstrated

the best overall performance. However, it did not significantly outperform the linear regression with binarization threshold resulted from the INPLS algorithm. Thus, application of INPLS without any additional classifier is preferable.

Chapter 6. BCI EXPERIMENTS RESULTS

6.1 Experiments in rats

Experimental data were collected during the behavioral experiments in a rat based on a simple reward-oriented task (see Chapter 2). Three series of behavioral experiments, 34 experiments in total (see Table 6.1), were collected over more than 8 months. The experiments lasted from 5 min up to 1 h (22 min in average). A part of one recording (#1 in Table 6.1) was used for calibration to identify the predictive model. The training data set included all event-related epochs and randomly selected 'non-event' epochs. The threshold of binarization was adjusted to every recording.

6.1.1 Calibration results

For calibration purposes, the signal of the training data set was mapped to the temporal–frequency–spatial space. Then, eight factors and the corresponding latent variables t_i, $i = \overline{1,8}$, were extracted by INPLS. All modalities of the first and the second factors are shown in Figure 6.1. Besides that, the predictive model for the pedal's position was constructed: $\hat{y} = \sum_{i=1}^{8} t_i b_i + b_0$, where b_0 is a parameter of centering. While INPLS extracts task-related factors, the influence of the factors is different. The coefficients b_i^*, of the normalized model $\hat{y} = \sum_{i=1}^{8} t_i^* b_i^* + b_0$, correspond to the weights of related factors in the final decomposition (see Section 4.6.3). These coefficients were $0.369, 0.273, 0.195, 0.087, 0.040, 0.030, 0.004, 0.002$ (Figure 6.2). The MI analysis revealed the leverages of each element for each modality (Figure 6.3). In particular, applied to the spatial modality, the MI analysis indicates occipital electrode number 15 as having the highest impact on the decision rule (~57% of the extracted information). High frequencies [100, 300] Hz provide the main contribution to the decision in the frequency modality (~86%). In the time domain the interval $[-0.5, 0]$ s before the event is the most significant (~68%).

To verify different frequency bands, the calibration procedure was applied separately for ECoG signal filtered in α- (10-15 Hz), β- (15-40 Hz), and γ- (55-300 Hz) bands. The relative weights of these bands in the final decomposition are: $\alpha: 0.38\%$, $\beta: 2.45\%$, $\gamma: 95.61\%$ (Figure 6.3). MI analysis applied to identified models revealed the leverages of elements of each modality and for every frequency band (Figure 6.4). In γ-band the most informative are: frequency ~147 Hz, moment 290 ms before event, and occipital electrode #15. In α-band the most informative are: frequency 10.5 Hz, moment 390 ms before event, and occipital electrodes #8 and #15. In β-band the most informative are: frequency ~33 Hz, occipital electrodes #8 and #15, the moment is rather defused and random (it could be explained by insignificance for BCI system of information contained in the β-band).

The temporal–frequency analysis for the best electrode (#15, over the cerebellar cortex) was the next step. Six factors were extracted by INPLS. The first three of them are shown in Figure 6.5. The relative weights of all six factors in the final decomposition are $0.568, 0.228, 0.102, 0.045, 0.042, 0.015$ (Figure 6.6).

Like in the case of all electrodes, the calibration procedure was applied separately for ECoG signal filtered in α- (10-15 Hz), β- (15-40 Hz), and γ- (55-300 Hz) bands for the electrode #15. MI analysis applied to identified models revealed the leverages of elements of each modality and for every frequency band (Figure 6.7). In γ-band the most informative are: frequency ~110 Hz, moment 290 ms before event. In α-band the most informative are: frequency 10.5 Hz, moment 410 ms before event. In β-band the most informative are: frequency ~33 Hz, like in the case of all electrodes the time moment is rather defused and random.

Comparison of the BCI performances (all electrodes versus single electrode) demonstrates that using of all electrodes does not significantly improve the approximation of the output variable on the training data set (Figure 6.8). The difference in RMSE is less than 2% for the number of factors ≥ 6. Projections of the points of observation into the first and the second factors (Figure 6.9) show good separability of the classes ‘event’ and ‘non-event’ in the single-electrode case even over almost eight months. From a computational point of view, the single-electrode predictive model was chosen. The calibration procedure resulted in the predictor of the pedal pressing events

and the threshold-based decision rule. The maximum of the OP criterion identifies the threshold balancing false activations and correct detections (Figure 6.10).

Then, the identified model was applied to predict the rat's control intention in the series of offline simulations as well as in the close-loop real-time BCI system.

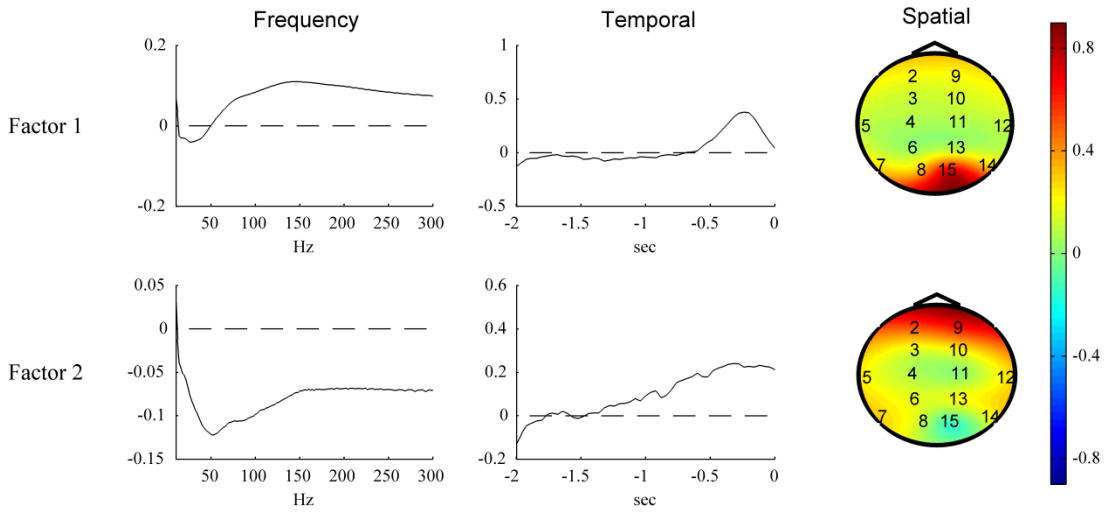

Figure 6.1 The first and the second factors (which are the most contributive out of eight): frequency, temporal and spatial projections; the values of elements of the spatial projector are shown in colors according to the color bar; positions of the electrodes are indicated by numbers.

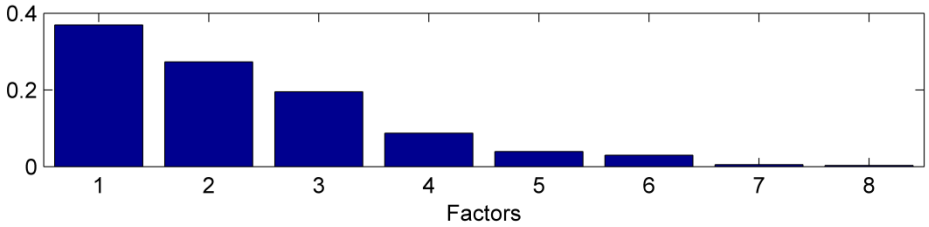

Figure 6.2 Factor's weights in the final decomposition.

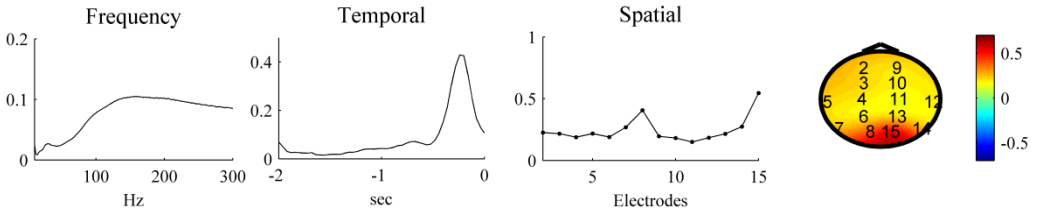

Figure 6.3 Impact on the predictive model of different modalities components according to MI analysis; the spatial modality is represented by the graph and the corresponding color map.

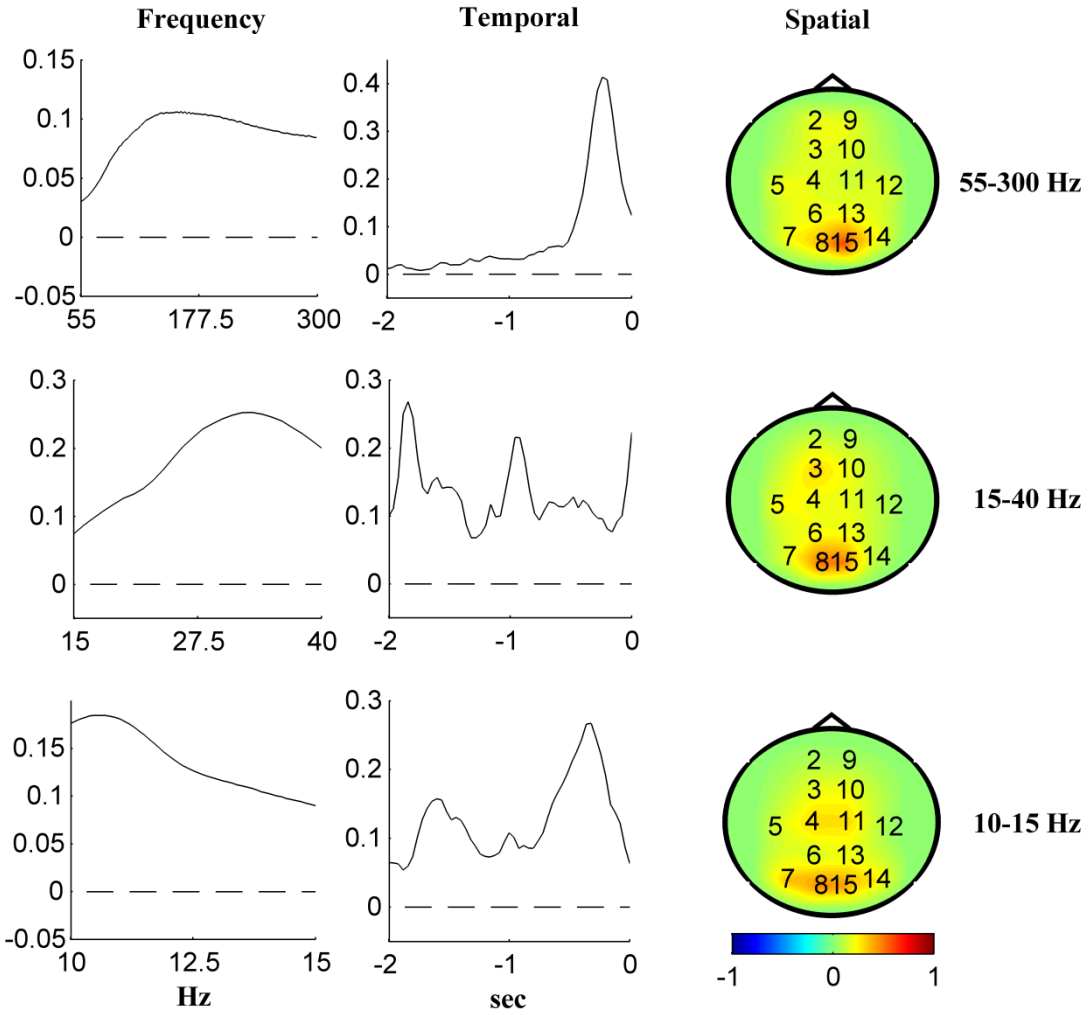

Figure 6.4 Impact on the predictive model of the components of different modalities according to the MI analysis; the spatial modality is represented by the graph and the corresponding color map. The predictive models are identified for three frequency bands: α-band (10-15 Hz), β-band (15-40 Hz), and γ-band (55-300 Hz).

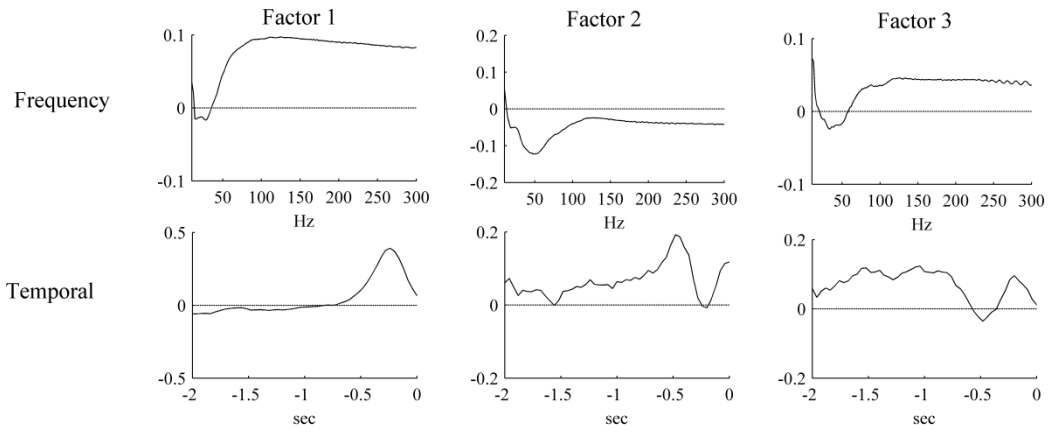

Figure 6.5 The best-electrode-calibration: frequency and temporal projections of the first three factors.

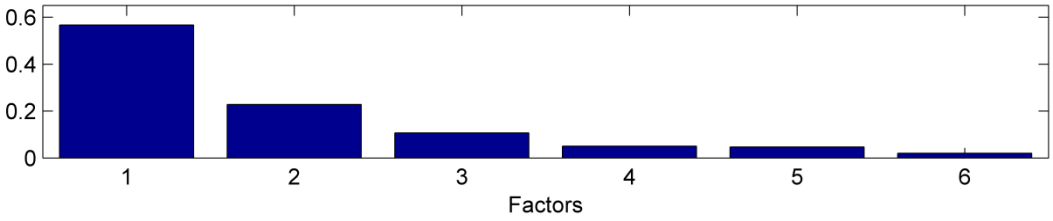

Figure 6.6 The best-electrode-calibration: weights of the factors in the final decomposition.

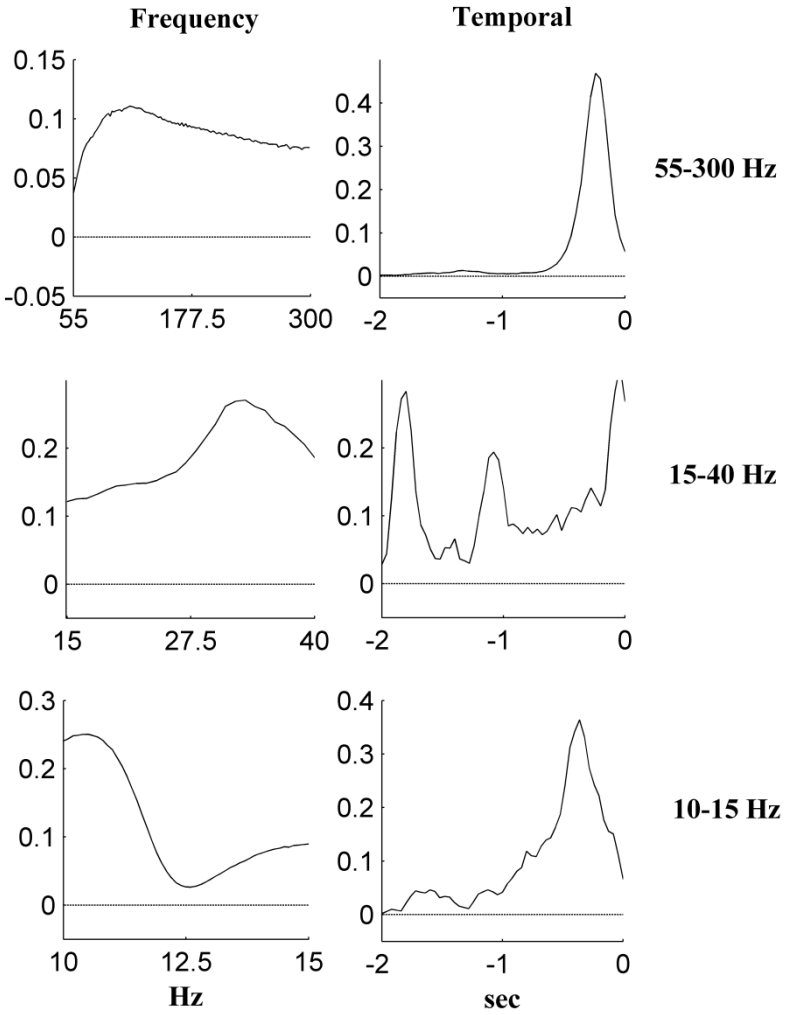

Figure 6.7 Impact on the predictive model of the components of different modalities according to the MI analysis. The predictive models are identified for three frequency bands: α-band (10-15 Hz), β-band (15-40 Hz), and γ-band (55-300 Hz).

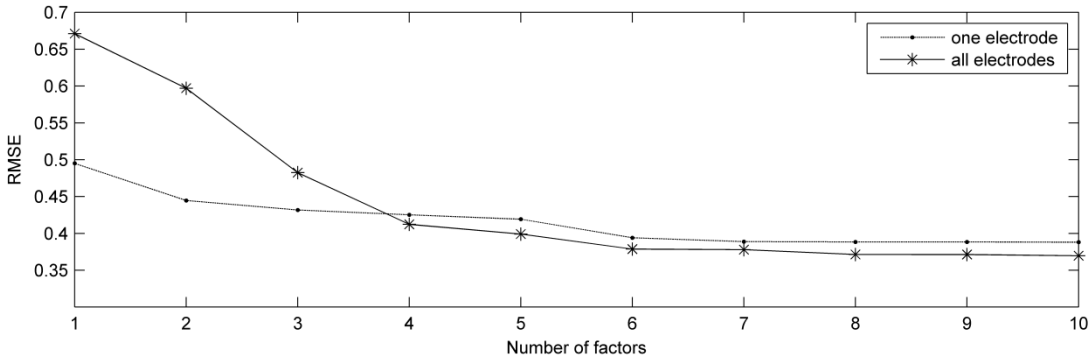

Figure 6.8 The best-electrode-calibration: comparison of the relative approximation errors, calculated on the training data set for the different numbers of factors: all electrodes versus the best electrode.

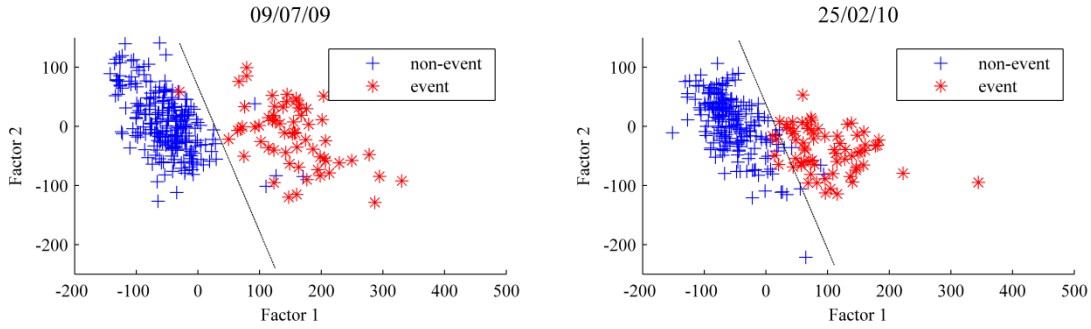

Figure 6.9 The best-electrode-calibration: projections of the 'event' and 'non-event' points of observations from the training recording to the first and the second factors. Even in eight months the quality of the separation remains considerable.

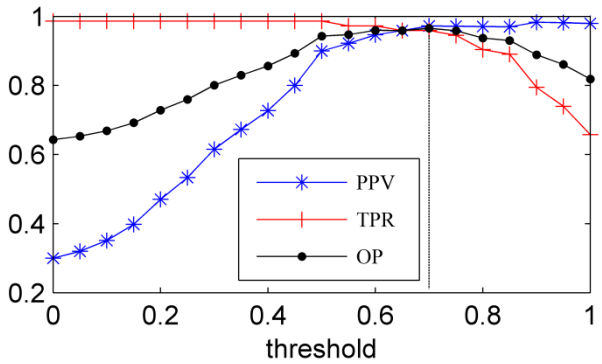

Figure 6.10 The best-electrode-calibration: the threshold of event detection is identifying according to the OP = (TPR+PPV)/2 criterion: TPR represents the rate of the detected events; (1–PPV) corresponds to the number of FP related to the number of system activations.

6.1.2 Offline and open-loop validation of the BCI system

For offline validation of the BCI system, the predictive model identified at the calibration stage is applied to the previously recorded data. This type of verification is preferable at the initial stages of the experiments when the animal is not trained enough and false activations of the distributor as well as misses of the pedal pushing significantly confuse the rat.

In the simulation mode, a decision about event/non-event state of the system was made every half a second buffer: the predictors were calculated each 0.125 s, i.e., 4 times per buffer. The buffer was considered as containing the 'event' if at least one of these predictors surpassed the threshold of binarization. After each detection, the system was blocked for 5 s to prevent multiple activations. Following to (Fatourechi et al., 2008) the real event was considered as detected (True Positive) if the time interval between the real event and its detection did not exceed 1.5 s.

For the first two groups of the BCI experiments (carried out in July and October, 2009; see from #1 to #12 in Table 6.1, Figure 6.11) the BCI performance and the delay of detection were evaluated. The mean overall performance OP_{sim} = 81.06% (79.65% excluding the training recording) was achieved with 0.51 false activations per minute (0.53 excluding the training record). Detections were made 0.18 s (on average) before the appearance of the real event. A fragment of one simulated experiment is shown in Figure 6.12. Delay time histograms for all these experiments are shown in Figure 6.13.

The open-loop experiments were carried out in the third session (the experiments of February and March, 2010; Table 6.1, Figure 6.11). In these experiments the rat obtained rewards after every pressing regardless decision of the predictive model. The answers generated by the system were analyzed independently afterwards. Like in the case of the offline validation, this type of the experiment was made to prevent the animal to be confused by false activation and pushing omissions. Experiments have demonstrated rather considerable decreasing of the quality: mean overall performance OP_{ol} = 50.34% was achieved with 1.46 false activations per minute. However, further examination of the rat's brain and the implanted electrodes, made after death of the animal, showed that this decreasing of the BCI performance was significantly caused by the electrodes degradation.

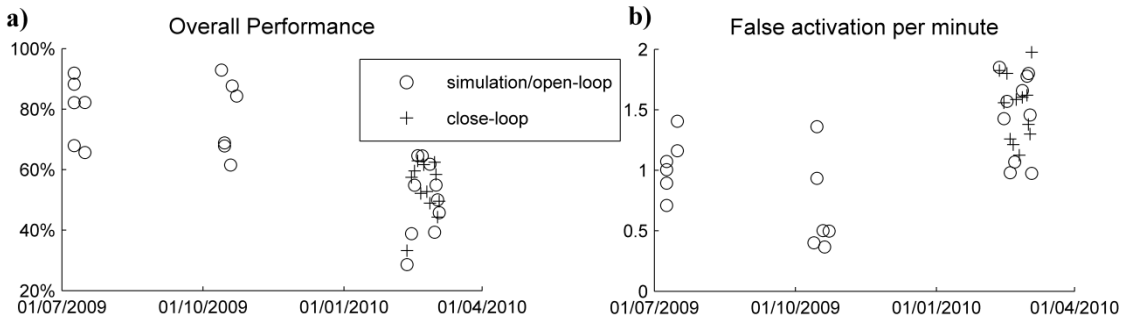

Figure 6.11 The Overall Performance (OP) and the number of false activation per minute calculated over 8 months (offline validation, open- and close-loop real-time experiments).

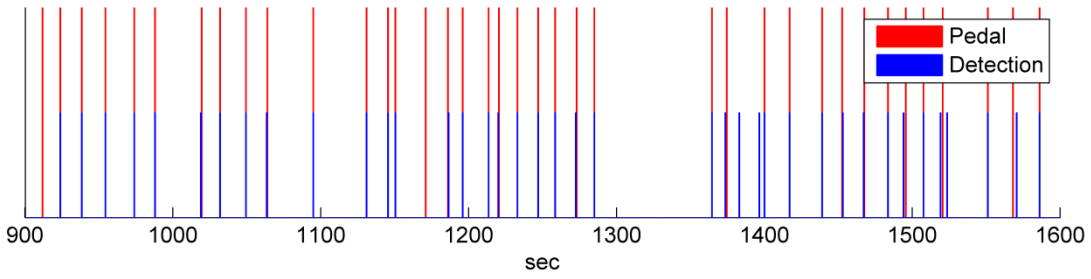

Figure 6.12 10-minutes-length fragment of the experiment #11 (from 20/10/09).

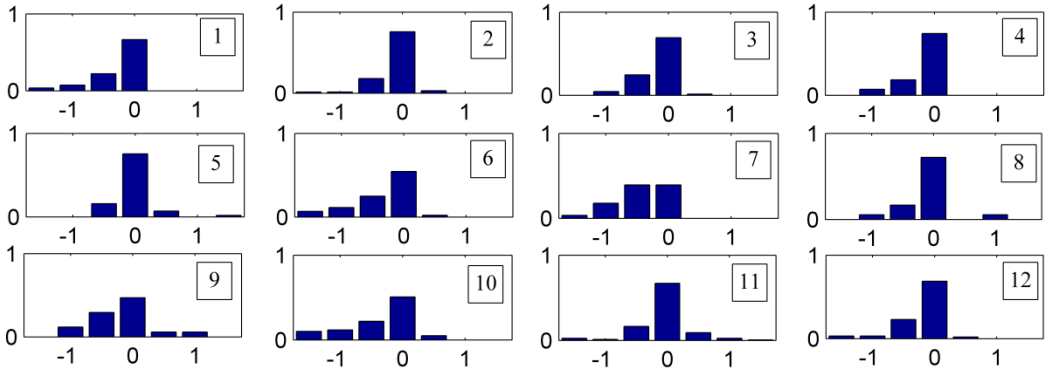

Figure 6.13 Time delay histograms of detection for the experiments from #1 to #12, zero moment corresponds to the real event time.

6.1.3 Close-loop real-time BCI experiments

In the close-loop real-time BCI experiments a food dispenser was controlled by the animal's brain activity. These tests were carried out during February and March,

2010 (Figure 6.11, Table 6.1). To clarify the organization of the experiments, some photos of one close-loop experiment are represented in Figure 6.14. Figure 6.15 demonstrates a 25 s length example of the close-loop detection made by the system in the real-time mode.

Like in the simulation mode and open-loop mode, the decision about event/non-event state of the system was made every half a second buffer (4 predictors per buffer, the buffer contained the 'event' if at least one of the predictors surpassed the threshold of binarization). The real event was considered as detected if distance between the real event and its detection did not exceed 1.5 s; the system was blocked for 5 s after the detection.

The mean overall performance of the close-loop experiments OP_{cl} = 53.61% with 1.52 false activations per minute. It should be especially emphasized that performances of the open-loop and the close-loop tests are similar at the same period of time (OP_{ol} = 50.34% and 1.46 false activations per minute, see Section 6.1.2). This conclusion has one significant practical consequence. Carrying out of the close-loop experiments in animals is connected with considerable difficulties: contrary to humans, the animals are very confused by the errors of the BCI system that could bring to impossibility of the experiment continuation. At the same time, the open-loop experiments could be easily implemented in animals without any considerable influence on the model's quality evaluation.

Another important requirement imposed on the BCI system is its possibility to function in the real-time conditions. Carried out experiments have confirmed satisfactory computation efficiency of the algorithm for the practical applications. The time for processing 0.5 second-buffer does not surpass 0.1 s (Intel Dual Core, 3.16 GHz; RAM 3.25 Gb). In addition, the evaluation experiments allow estimation of the response time of software that is on average 0.31 s including data collecting and data processing time: 0.25 and 0.06 s, respectively.

Figure 6.14 The close-loop real-time stage of the experiment. The rat presses the pedal but decision whether to give a reward is made on the basis of the ECoG signal.

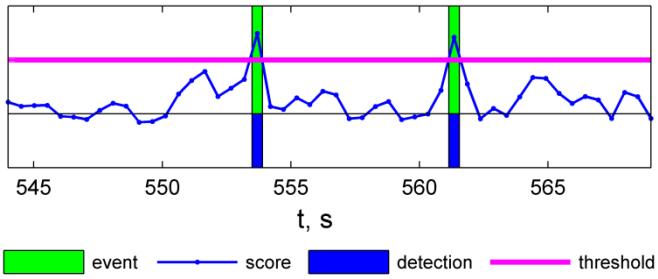

Figure 6.15 25 second-length example of the close-loop BCI experiment: two real events were detected by the system.

Table 6.1 Performances of both open-loop and close-loop BCI experiments.

No	Date	No. of events	Time (s)	FP	FN	TP	TN	TPR (%)	PPV (%)	FPR (%)	ERR (%)	OP (%)	FP min^{-1}	Conditions
1[a]	09.07.2009	73	592	2	3	70	1109	95,89	97,22	0,18	0,42	96,56	0,20	s
2	09.07.2009	74	895	9	13	61	1707	82,43	87,14	0,52	1,23	84,79	0,60	s
3	09.07.2009	56	895	11	24	32	1723	57,14	74,42	0,63	1,96	65,78	0,74	s
4	09.07.2009	68	537	6	11	57	1000	83,82	90,48	0,60	1,58	87,15	0,67	s
5	16.07.2009	37	310	0	7	30	583	81,08	100,00	0,00	1,13	90,54	0,00	s
6	16.07.2009	33	427	3	6	27	818	81,82	90,00	0,37	1,05	85,91	0,42	s
7	13.10.2009	49	599	3	5	44	1146	89,80	93,62	0,26	0,67	91,71	0,30	s
8	15.10.2009	46	964	15	18	28	1867	60,87	65,12	0,80	1,71	62,99	0,93	s
9	15.10.2009	28	397	6	10	18	760	64,29	75,00	0,78	2,02	69,64	0,91	s
10	19.10.2009	99	3587	32	40	59	7043	59,60	64,84	0,45	1,00	62,22	0,54	s
11	20.10.2009	169	3600	18	17	152	7013	89,94	89,41	0,26	0,49	89,68	0,30	s
12	23.10.2009	114	1808	14	18	96	3488	84,21	87,27	0,40	0,88	85,74	0,46	s
13	19.02.2010	33	2400	73	17	16	4694	48,48	17,98	1,53	1,88	33,23	1,83	c
14	19.02.2010	37	940	29	26	11	1814	29,73	27,50	1,57	2,93	28,61	1,85	o
15	22.02.2010	30	385	10	14	16	730	53,33	61,54	1,35	3,12	57,44	1,56	c
16	22.02.2010	33	715	17	21	12	1380	36,36	41,38	1,22	2,66	38,87	1,43	o
17	23.02.2010	45	800	24	16	29	1531	64,44	54,72	1,54	2,50	59,58	1,80	c

[a]This recording was used for calibration.

s – simulation mode, o – open-loop mode, c – close-loop mode.

Table 6.1 Performances of both open-loop and close-loop BCI experiments.

No	Date	No. of events	Time (s)	FP	FN	TP	TN	TPR (%)	PPV (%)	FPR (%)	ERR (%)	OP (%)	FP min^{-1}	Conditions
18	23.02.2010	77	2050	43	23	54	3980	70,13	55,67	1,07	1,61	62,90	1,26	c
19	23.02.2010	31	535	14	14	17	1025	54,84	54,84	1,35	2,62	54,84	1,57	o
20	24.02.2010	112	1980	40	59	53	3808	47,32	56,99	1,04	2,50	52,16	1,21	c
21	24.02.2010	30	490	8	12	18	942	60,00	69,23	0,84	2,04	64,62	0,98	o
22	25.02.2010	152	2500	66	55	97	4782	63,82	59,51	1,36	2,42	61,66	1,58	c
23	25.02.2010	31	505	9	12	19	970	61,29	67,86	0,92	2,08	64,57	1,07	o
24	26.02.2010	28	480	9	15	13	923	46,43	59,09	0,97	2,50	52,76	1,13	c
25	26.02.2010	115	2170	58	59	56	4167	48,70	49,12	1,37	2,70	48,91	1,60	c
26	26.02.2010	39	615	17	14	25	1174	64,10	59,52	1,43	2,52	61,81	1,66	o
27	01.03.2010	180	3000	81	62	118	5739	65,56	59,30	1,39	2,38	62,43	1,62	c
28	01.03.2010	29	540	16	18	11	1035	37,93	40,74	1,52	3,15	39,34	1,78	o
29	02.03.2010	163	2350	54	74	89	4483	54,60	62,24	1,19	2,72	58,42	1,38	c
30	02.03.2010	30	500	15	13	17	955	56,67	53,13	1,55	2,80	54,90	1,80	o
31	03.03.2010	247	3600	78	159	88	6875	35,63	53,01	1,12	3,29	44,32	1,30	c
32	03.03.2010	31	700	17	15	16	1352	51,61	48,48	1,24	2,29	50,05	1,46	o
33	04.03.2010	140	2400	79	68	72	4581	51,43	47,68	1,70	3,06	49,56	1,98	c
34	04.03.2010	59	800	13	40	19	1528	32,20	59,38	0,84	3,31	45,79	0,98	o

s – simulation mode, o – open-loop mode, c – close-loop mode.

6.1.4 Validation of the RNPLS algorithm with real data

The RNPLS algorithm was tested with the real data sets which were collected during the BCI experiments in freely moving rat. The data were extracted from the record #1 (see Table 6.1). 1000 randomly selected points (700 correspond to "non-events" and 300 to "events") formed the training set, whereas 400 points (300+100) were used as the test set. The intersection of the training and the test data sets was empty. The NPLS algorithm was trained on the whole training set. For the RNPLS the training set was split into disjoint subsets with 10 and 100 points. Then the projectors and the predictive models were identified. Figure 6.16 represents the first two factors (the total number of factors was taken equal to 5 by cross-validation procedure) calculated by the RNPLS (10). The relative weights of the factors in the final decomposition are demonstrated in Figure 6.17. Figure 6.18 shows the percentage of the resulting prediction errors. With respect to NPLS, the RNPLS algorithm demonstrates minimal deterioration in the prediction quality: for RNPLS (100) it is about 0.1%, whereas for RNPLS (10) it is about 0.2%.

Thus, the proposed recursive algorithm demonstrated the excellent performance in comparison to the NPLS approach in terms of accuracy and convergence rate on the real data. At the same time, taking into account that requirements for computation resources (memory) are low and do not depend on size of the processed data, the method is favorable for the BCI application.

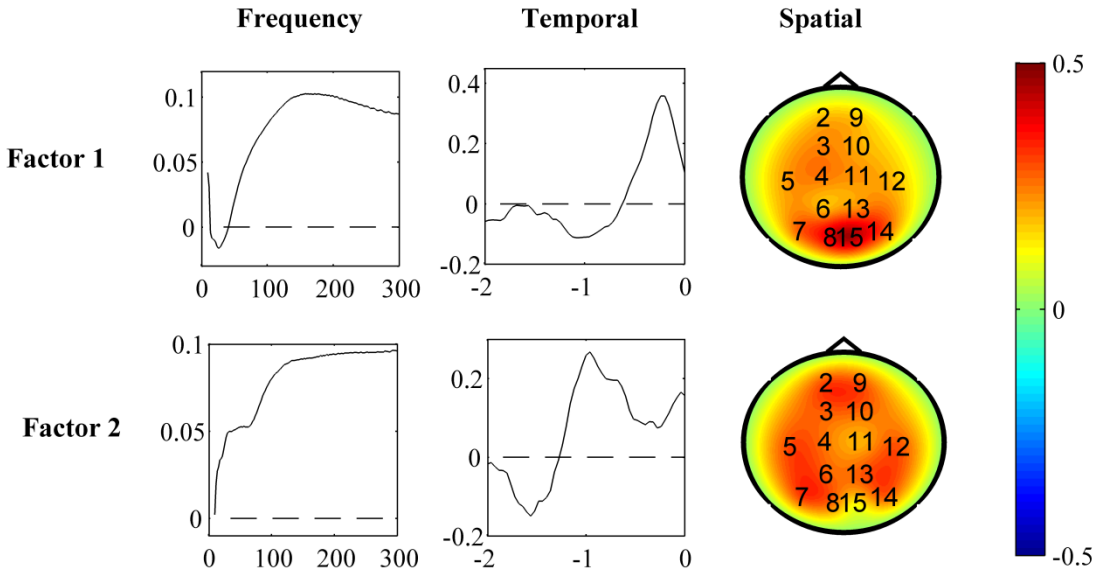

Figure 6.16 Frequency, temporal, and spatial projectors of the first and the second factors identified by RNPLS (10).

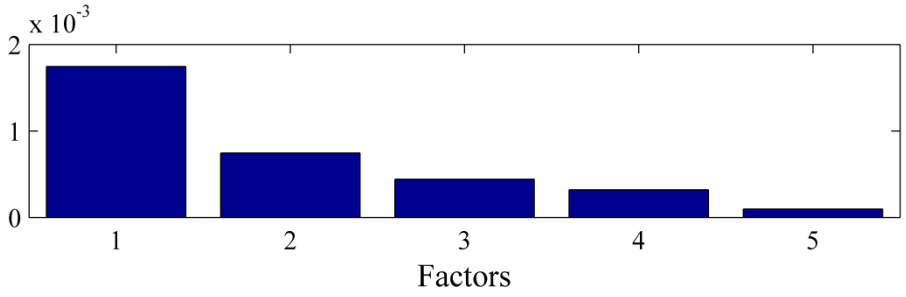

Figure 6.17 Weights of the factors in the final decomposition (evaluated by the RNPLS (10) algorithm).

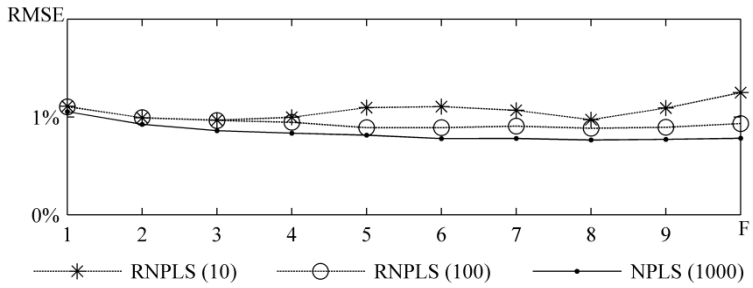

Figure 6.18 Comparison of the test data prediction error (RMSE). RNPLS (10) – the training set is split into 10-points disjoint subsets; RNPLS (100) – the training set is split into 100-points disjoint subsets, NPLS (1000) – generic NPLS using the whole training set.

6.2 Experiments in monkeys

Experimental data were collected during the behavioral experiments based on a simple reward-oriented task (see Chapter 2). The model was applied to the data recorded in 24 experiments (see Table 6.2, Figure 6.25). The experiments lasted from 4 min up to 20 min, 8 min in average. Parts of four recordings were used for the calibration that is to identify the predictive models. The training data sets included all event-related epochs and randomly selected 'non-event' epochs. The threshold of binarization was adjusted to each recording.

6.2.1 Results of the calibration

The monkey's brain activity signal of the training data set was mapped to the temporal–frequency–spatial space for the calibration purposes. Then, five factors (the number is defined by the cross-validation procedure) and the corresponding latent variables t_i, $i = \overline{1,5}$, were extracted by the L1-Penalized NPLS algorithms (see Section 4.7, $\lambda = 0.9\lambda_{max}$) separately for all four positions of the lever. The algorithm was chosen taking into account computation restrictions of the real-time model application to find a subset of electrodes mostly participating in the final model. The coefficients b_i^*, of the normalized predictive model $\hat{y} = \sum_{i=1}^{5} t_i^* b_i^* + b_0$, correspond to the weights of the related factors in the final decomposition (see Section 4.6.3). These coefficients are represented in Figure 6.19 for all positions of the pedal. The values of the coefficients are:

‘left’: $0.346, 0.273, 0.232, 0.111, 0.038$;

‘right’: $0.346, 0.217, 0.195, 0.138, 0.104$;

‘up’: $0.383, 0.263, 0.158, 0.151, 0.045$;

‘down’: $0.278, 0.210, 0.194, 0.182, 0.138$.

All modalities of the first factor are shown in Figure 6.20 for each position of the pedal. The MI analysis revealed the leverages of elements of each modality (Figure 6.21).

Applied to the spatial modality, the MI analysis indicates that the electrode #22 located in the motor primary cortex has the highest impact on the decision rule (84%, 97%, 89%, and 75% of extracted information for ‘left’, ‘right’, ‘up’, and ‘down’ positions of the pedal, respectively). High frequencies (≥ 100 Hz) make significant contribution to the decision in the frequency modality, however, contrary to the rat, the influence of the low frequencies (< 100 Hz) is also rather considerable, especially, in the case of the ‘left’ position of the pedal. In the time domain the interval $[-0.2, 0]$ s before the event is the most significant for all positions of the lever. The temporal–frequency analysis for the best electrode (#22) was the next step. Seven factors were extracted by INPLS. The first one of them is shown in Figure 6.22 for every position of the pedal. The relative weights of all seven factors in the final decomposition are (Figure 6.23):

‘left’: $0.419, 0.234, 0.116, 0.092, 0.089, 0.039, 0.011$;

‘right’: $0.371, 0.228, 0.137, 0.122, 0.079, 0.043, 0.020$;

‘up’: $0.313, 0.201, 0.182, 0.137, 0.092, 0.073, 0.002$;

‘down’: $0.232, 0.212, 0.167, 0.121, 0.106, 0.082, 0.080$.

The MI analysis revealed the leverages of elements of each modality for the best electrode case (Figure 6.24). Like in the case of the rat, taking into account the computational cost, the single-electrode predictive model was chosen for the BCI system.

The calibration procedure results in the predictor of the pressing events and the threshold-based decision rule. Calibration-based model was applied for prediction of the animal’s control intention in the series of offline simulation BCI experiments.

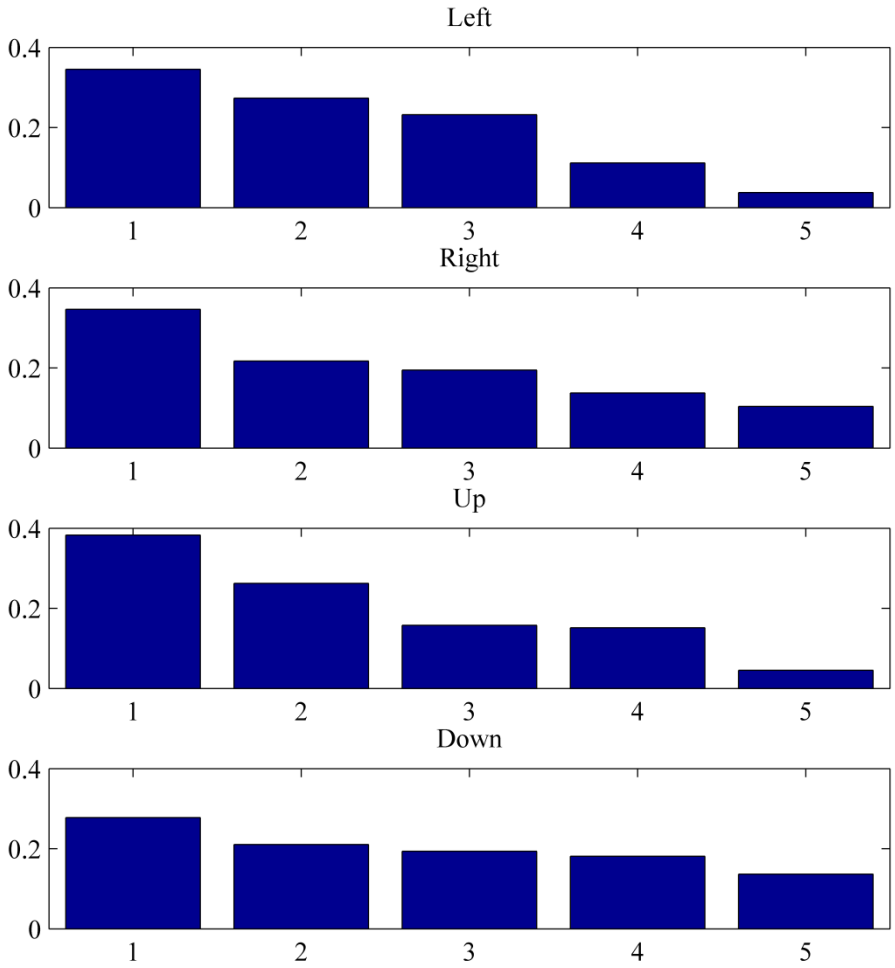

Figure 6.19 Factors weights in the final decomposition for every position of the pedal.

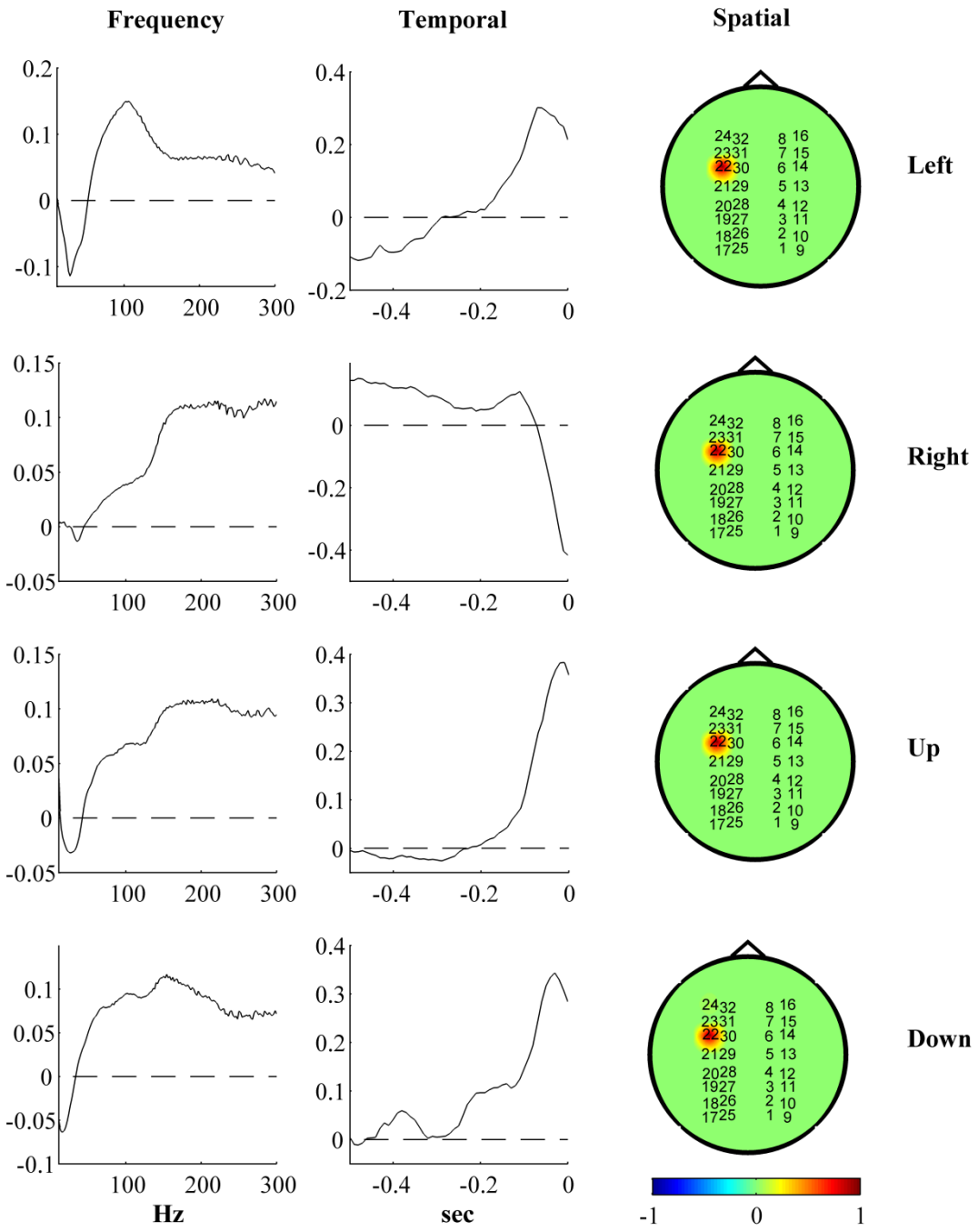

Figure 6.20 The first factor (according to the influence on the final decision): frequency, temporal and spatial projections for every position of the pedal. The values of elements of the spatial projectors are shown in colors according to the color bar; the electrodes positions are indicated by numbers.

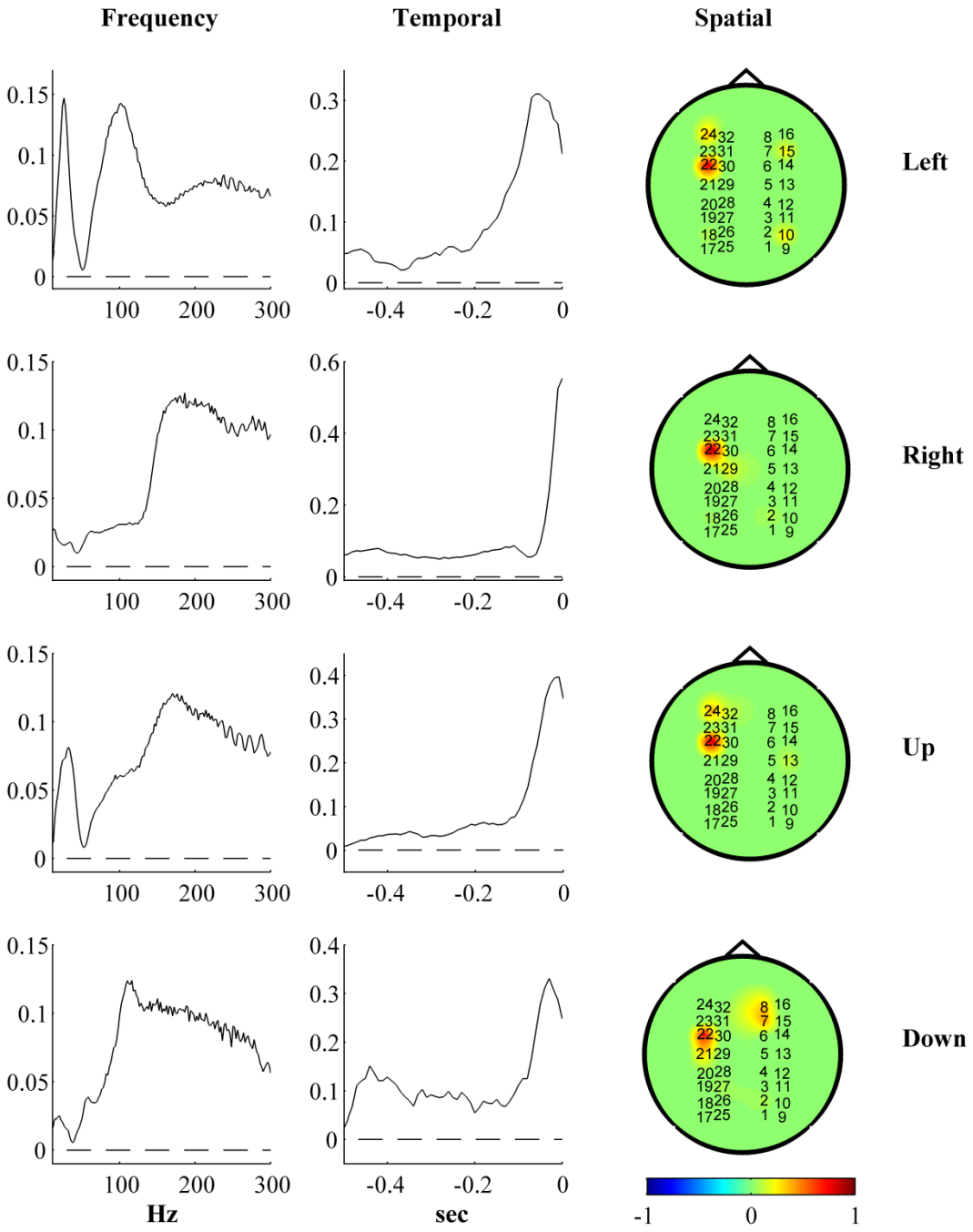

Figure 6.21 Impact on the predictive model of the components of different modalities according to the MI analysis of the LPNPLS predictive models for each pedal position; the spatial modalities are represented by the graphs and the corresponding color map.

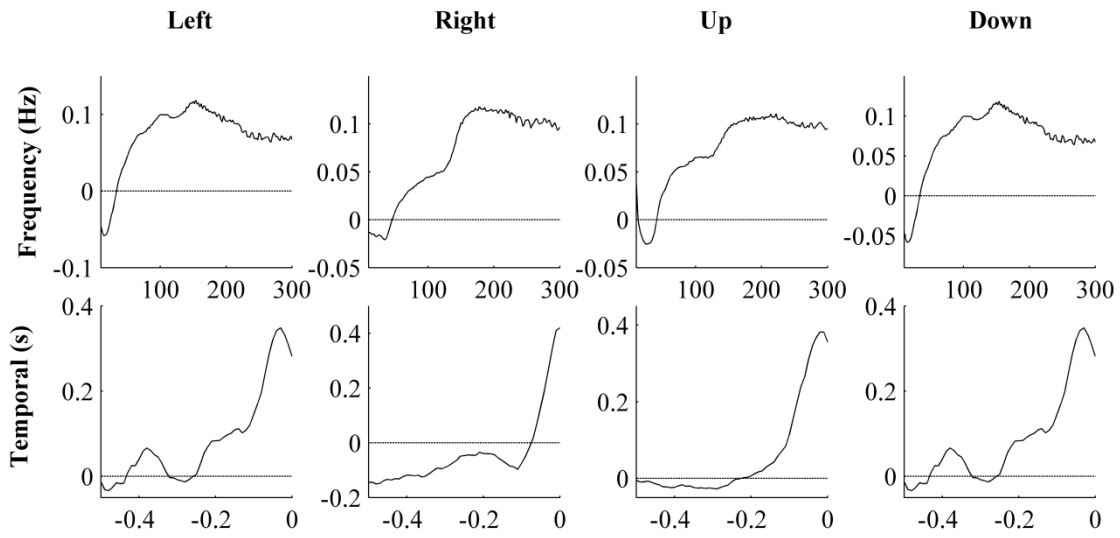

Figure 6.22 The best-electrode-calibration: frequency and temporal projections of the first factors for each position of the pedal.

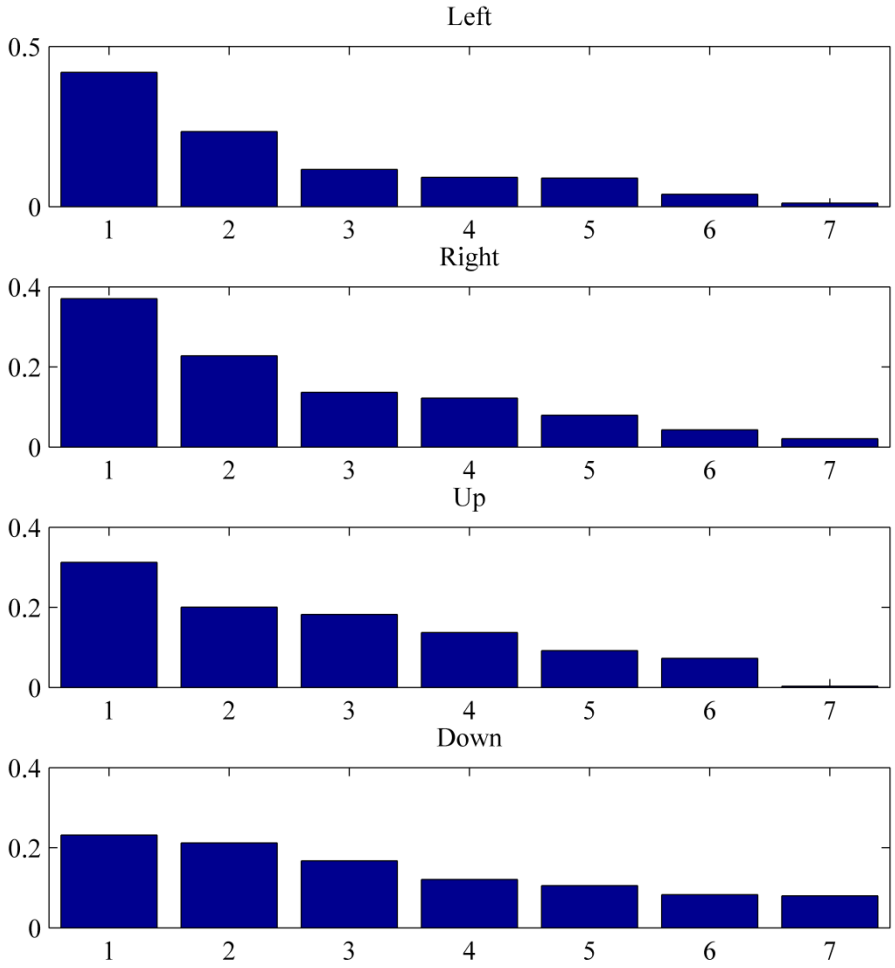

Figure 6.23 The best-electrode-calibration: the factors weights in the final decomposition for each position of the pedal.

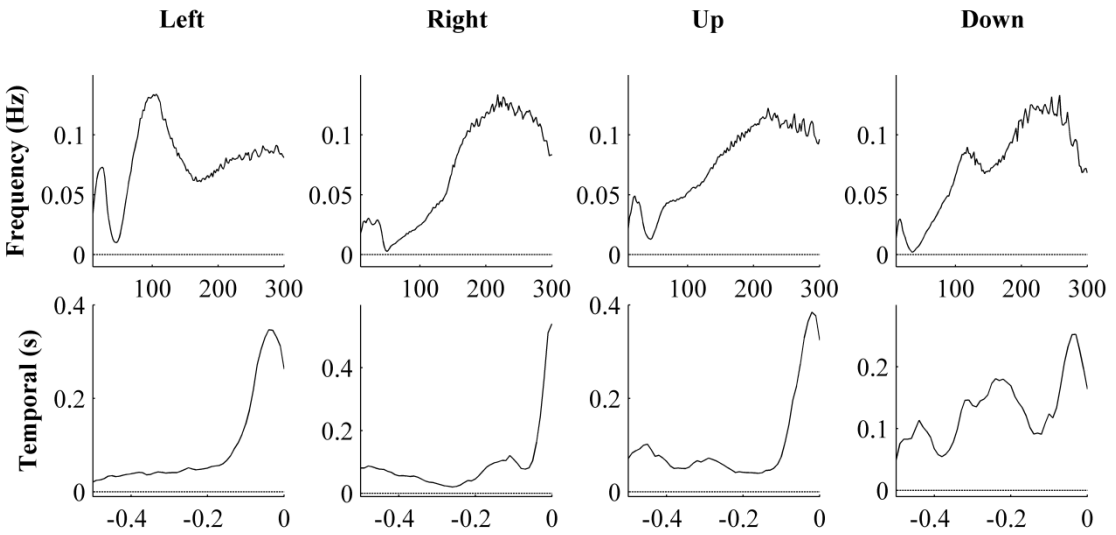

Figure 6.24 Impact on the predictive model of the components of the different modalities according to the MI analysis for each pedal position.

6.2.2 Offline validation of the BCI system

The event prediction procedure was validated in a set of simulations of the real-time experiments. A decision was made every 0.5 s buffer: the predictors were calculated each 0.125 sec. The buffer was considered as containing an 'event' if at least one of these predictors surpassed the threshold of binarization. After each detection, the system was blocked for 1.5 s to prevent multiple activations, the real event was considered as detected if the time interval between the real event and its detection did not exceed 1.5 s.

To make calibration of the BCI system and assess performance of the predictive models 24 experiments (from 4 to 20 min, 8 min in average) were carried out (see Table 6.2, Figure 6.25). Four recordings were used for calibration. Identified models were tested in all 24 recordings.

Table 6.2 Performance of the BCI experiments.

	No	Date	No. of events	Time (s)	FP	FN	TP	TN	TPR (%)	PPV (%)	FPR (%)	ERR (%)	OP (%)	FP min^{-1}
Left	1	10.03.2011	85	468,5	29	27	58	823	68,24	66,67	3,40	5,98	67,45	3,71
	2[a]	14.03.2011	109	845,5	32	31	78	1550	71,56	70,91	2,02	3,73	71,23	2,27
	3	18.03.2011	83	347	23	24	59	588	71,08	71,95	3,76	6,77	71,52	3,98
	4	24.03.2011	109	506	32	39	70	871	64,22	68,63	3,54	7,02	66,42	3,79
	5	14.04.2011	83	515	30	30	53	917	63,86	63,86	3,17	5,83	63,86	3,50
	6	19.04.2011	19	276	5	9	10	528	52,63	66,67	0,94	2,54	59,65	1,09
Right	7	10.03.2011	82	498,5	36	27	55	879	67,07	60,44	3,93	6,32	63,76	4,33
	8	11.03.2011	94	1107	65	55	39	2055	41,49	37,50	3,07	5,42	39,49	3,52
	9	30.03.2011	62	283	12	10	52	492	83,87	81,25	2,38	3,89	82,56	2,54
	10	04.04.2011	85	304,5	32	33	52	492	61,18	61,90	6,11	10,67	61,54	6,31
	11[a]	14.04.2011	88	227,5	9	10	78	358	88,64	89,66	2,45	4,18	89,15	2,37
	12	18.04.2011	78	353,5	26	20	58	603	74,36	69,05	4,13	6,51	71,70	4,41
Up	13[a]	15.03.2011	104	535	27	31	73	939	70,19	73,00	2,80	5,42	71,60	3,03
	14	17.03.2011	69	672	29	35	34	1246	49,28	53,97	2,27	4,76	51,62	2,59
	15	24.03.2011	32	336	14	8	24	626	75,00	63,16	2,19	3,27	69,08	2,50
	16	25.03.2011	63	380,5	23	26	37	675	58,73	61,67	3,30	6,44	60,20	3,63
	17	30.03.2011	90	339,5	21	26	64	568	71,11	75,29	3,57	6,92	73,20	3,71
	18	14.04.2011	70	361,5	11	23	47	642	67,14	81,03	1,68	4,70	74,09	1,83
Down	19	10.03.2011	38	521,5	27	29	9	978	23,68	25,00	2,69	5,37	24,34	3,11
	20	28.03.2011	85	824,5	45	42	43	1519	50,59	48,86	2,88	5,28	49,73	3,27
	21[a]	30.03.2011	53	322,5	20	20	33	572	62,26	62,26	3,38	6,20	62,26	3,72
	22	14.04.2011	47	341,5	17	18	29	619	61,70	63,04	2,67	5,12	62,37	2,99
	23	18.04.2011	52	516	26	29	23	954	44,23	46,94	2,65	5,33	45,58	3,02
	24	19.04.2011	76	390,5	27	34	42	678	55,26	60,87	3,83	7,81	58,07	4,15

[a]These recordings are used for calibration.

The means of the performance characteristics are (values computed without the training records are given in the brackets):

left: OP = 66.69±4.52% (65.68±4.40%), FP/min = 3.06±1.14 (3.21±1.20),

right: OP = 68.03±17.58% (73.74±11.91%), FP/min = 3.92±1.45 (3.99±1.61),

up: OP = 66.63±8.90% (69.63±5.61%), FP/min = 2.88±0.72 (2.94±0.79),

down: OP = 50.39±14.46% (50.53±16.16%), FP/min = 3.38±0.46 (3.40±0.52).

A fragment of one simulated experiment as well as its time-delay histogram are shown in Figure 6.26. In average, the event prediction was made 0.04 s before the event. Figure 6.27 demonstrates a set of photos of the real-time experiment.

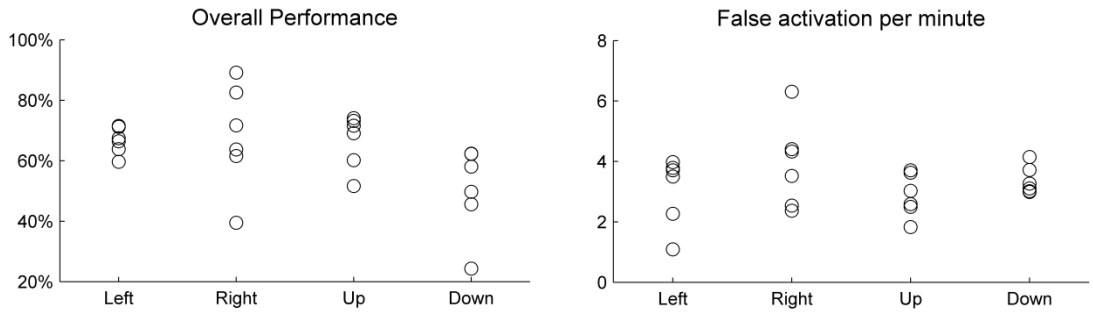

Figure 6.25 The Overall Performance (OP) and the number of false activations per minute obtained in the experiments for each pedal position.

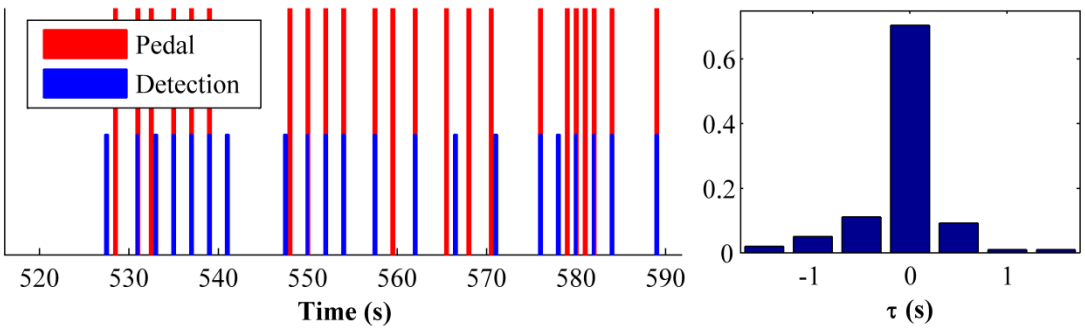

Figure 6.26 1.5 min length fragment of the experiment #2 (from 14/03/2011) and the time-delay histogram of detections.

Figure 6.27 Photos of the real-time stage of the experiment.

6.2.3 Validation of the LPNPLS algorithm with real data

The L1-Penalized NPLS (LPNPLS) method was tested with real data collected during the BCI experiments in monkey. For validation of the LPNPLS algorithm it was compared with generic NPLS. Four files were used for calibration (#2 for 'left', #11 for 'right', #13 for 'up', and #21 for 'down' pedal positions, see Table 6.2, Section 6.2.1). The factors number in the predictive models ranges from 1 to 5. Identified models were applied to the test files (one file for each pedal position).

The computational experiments demonstrated that results of L1-Penalized NPLS are comparable with NPLS ones or outperform them in terms of root mean squared error (RMSE). The comparison was carried out for all positions of the pedal and number of factors ranging from 1 to 5 (Figure 6.28). At the same time, it should be emphasized that the LPNPLS algorithm identifies the sparse predictive model. Contrary to the NPLS approach, only few electrodes are used for decision making (6, 6, 7, and 9 for 'left', 'right', 'up', and 'down' positions of the pedal, respectively). Results of the MI analysis are represented for each pedal position in Figure 6.21 for LPNPLS, and in Figure 6.29 for NPLS (number of factors $F = 5$). Thus, the Penalized NPLS algorithm effectively reduces the number of utilized electrodes without loss of prediction quality. Moreover, due to the reduction of the feature space, a considerable prediction improvement can be achieved.

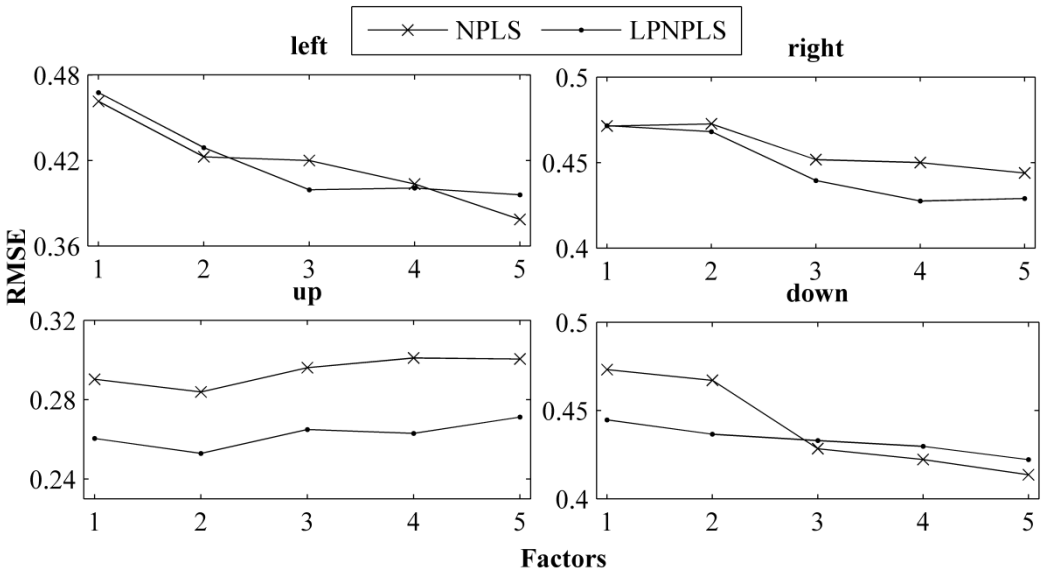

Figure 6.28 Comparison of prediction errors (root mean squared error, RMSE $(\sum_k |y_k - \hat{y}_k|^2)^{0.5}$) for the NPLS and the LPNPLS algorithms on the test set for different number of factors and different pedal positions.

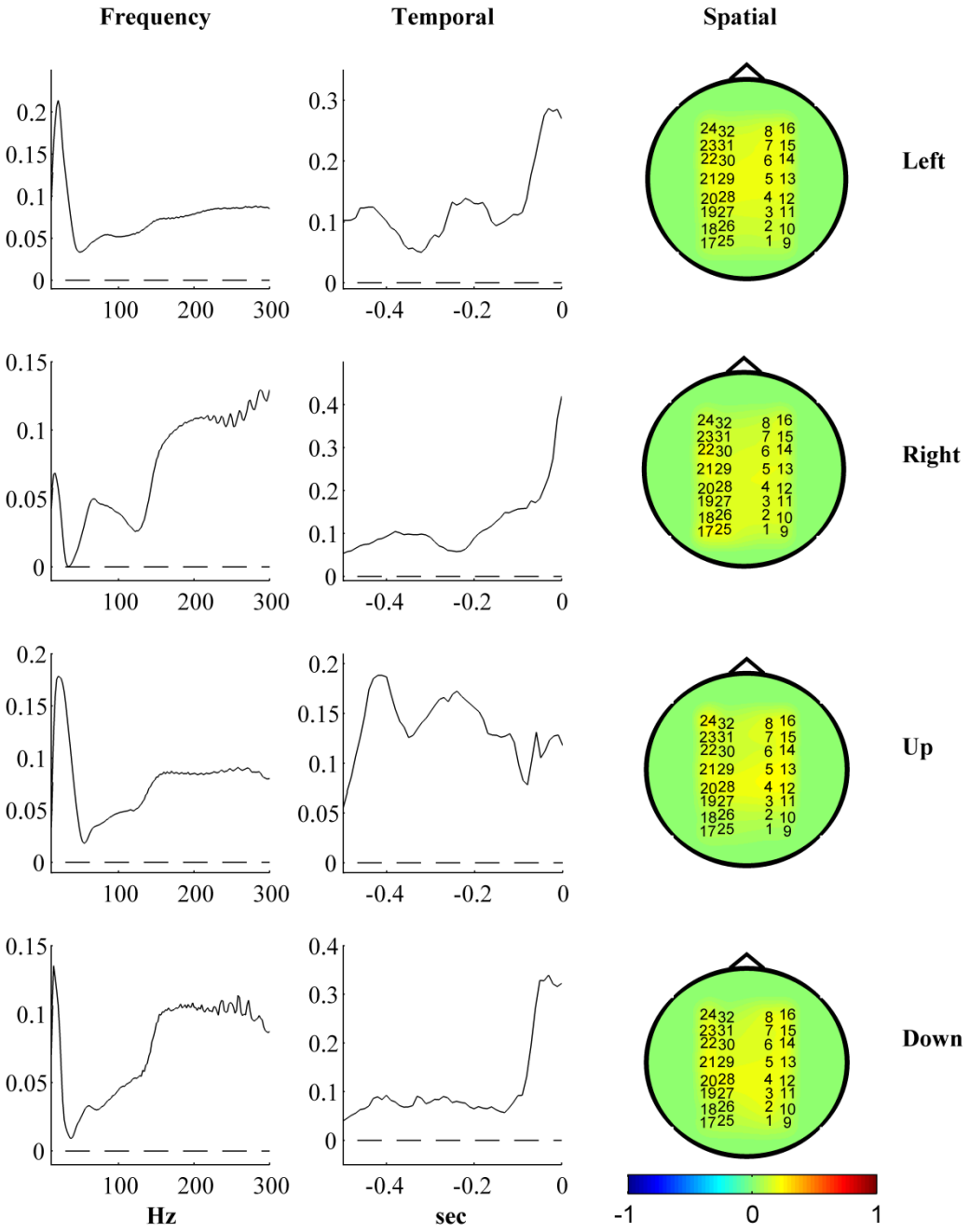

Figure 6.29 Impact on the predictive model of the components of different modalities according to the MI analysis of the NPLS predictive models for each pedal position; the spatial modalities are represented by the graphs and the corresponding color map.

6.3 Discussion

6.3.1 Experiments in rodent

The experiments in rodents have demonstrated that the most posterior electrodes have the strongest influence on the resulting model. These electrodes are located in the region of cerebellum. Additional experiments in several animals will allow better studying the location of zone of interest, its stability, evolution over time, neuroplasticity, etc.

The identified factors can be interpreted taking into account their influence on the final model. The first factor is responsible for almost 57% of extracted information (see Section 6.1.1). For this factor the weights in the frequency-modality are positive for the low and high frequencies (α- and γ-bands: 10-15 and 55-300 Hz), and negative for β-band [15-40] Hz. The signs of weights can be resulted from the changes of the signal energy in the corresponding frequency bands. The decrease of signal intensity in the β-band could be associated with a motor related potential (MRCP) (e.g., Nascimento et al., 2006; Boye et al., 2008; Gu et al., 2009; Cabrera et al., 2010). At the same time, the MI analysis demonstrates that high frequencies (>100 Hz) have the most significant influence on the decisions. However, in the low frequency (~10.5 Hz) the event-related activity appears approximately 100 ms before the activity in γ-band. In parallel, the MI analysis determined one electrode in the occipital cortex which has the greatest influence on the decision. It was demonstrated that the detection model, using only this electrode, has comparable prediction quality with the one, using all electrodes. Thus, taking into account the question of computational efficiency, the one-electrode predictive model seems preferable for the future implementation in the real-time BCI system.

While the training data set in the rat's experiments was composed from 10-minutes recording (about 1.3% of all available data), the observations of more than 750 minutes (12.5 hours) were used for test. The same predictive model was applied in all tests. The only adjusted parameter was the detection threshold. It was slightly varying daily, that could be explained by instability of the acquisition chain as well as by variability of the brain activity. This parameter remains stable during the day of experiment. It can be adjusted from the first several minutes (3-5 min) of the

experimental session of the real-time BCI. Three groups of experiments were carried out over almost eight months. In the first two groups of experiments (simulation mode, Table 6.1) about 80% of the correct event detections were achieved (TPR = 78±13% in average), while the level of the false activations was acceptably low (FPR = 0.44±0.24% in average). In almost eight months after the calibration model was obtained (the last group of the experiments in Table 6.1), TPR = 48±13%, FPR = 1.25±0.29% in the open-loop and TPR = 54±10%, FPR = 1.30±0.23% in the close-loop experiments. The deterioration of the system's performance could be explained by significant degradation of electrodes. At the same time, it should be mentioned that both the open- and the close-loop tests gave similar results. Moreover, at least part of the false positives of all the groups of experiments resulted from artifacts produced along the recording chain (connectors and so on) and can be eliminated using wireless data transferring (Charvet et al., 2011). However, additional experiments in several animals are necessary to make more reliable conclusions. Finally, carried out close-loop real-time experiments have demonstrated efficiency of implementation of the proposed method in the real BCI systems. Moreover, data collecting and processing time (for a 0.5 s buffer of data: 0.25 s and 0.06 s, respectively) allows application of the proposed approach for several degrees of freedom.

6.3.2 Experiments in monkey

The experiments in the monkey were carried out to verify the proposed approaches in other animals. The experiments have shown that contrary to the rat, the strongest influence on the resulting model have electrodes located in the motor primary cortex. However, to determine a long-term stability, time evolution, neuroplasticity, etc., additional experiments should be carried out.

The one-electrode model calibration was made on the basis of data from the primary motor cortex electrode, determined by the L1-Penalized NPLS algorithm. The MI analysis applied to the identified model has shown that, similar to the case of the rat, the low frequency band (<45 Hz) had relatively small influence on decision, whereas the high frequencies (>100 Hz) contain the most significant data. Thus, although the informative electrodes are located in the different parts of the rat's and monkey's brains

(cerebellum region in the case of rat, and motor primary cortex in the case of monkey), the identified activity responsible for the pedal-pressing events is rather similar.

The training data set in the monkey's experiments were composed from four recordings (30-minutes length in total, which is about 17% of all available data), the observations of more than 180 minutes (3 hours) duration were used for testing. The only adjusted parameter was the detection threshold. Like in the case of the rat, it was slightly varying daily. During the tests, up to 70% of the correct event detections were achieved in average ($TPR_{left}^{average} = 65 \pm 7\%$, $TPR_{right}^{average} = 69 \pm 17\%$, $TPR_{up}^{average} = 65 \pm 10\%$, and $TPR_{down}^{average} = 50 \pm 14\%$), while the average level of the false activations was acceptably low ($FPR_{left}^{average} = 2.8 \pm 1.1\%$, $FPR_{right}^{average} = 3.7 \pm 1.4\%$, $FPR_{up}^{average} = 2.6 \pm 0.7\%$, and $FPR_{down}^{average} = 3.0 \pm 0.5\%$). Some part of the false positives can be also eliminated by means of the wireless data transferring. Additional experiments should be carried out to make more reliable conclusions about other possible reasons of the quality deterioration.

Chapter 7. CONCLUSIONS

7.1 Summary

This study was undertaken as a step toward the fully autonomous (self-paced) BCI functioning in natural environment which is of crucial importance for efficient BCI clinical applications. The self-paced BCI in noisy natural environment requires a high level of selectivity for identification and discrimination of the specific neuronal activity against the background brain functioning during continuous monitoring. To approach this goal, we have studied the binary self-paced BCI in animals (rats freely moving in the cage and monkeys sitting on chair). Namely, we have analyzed the recordings of the series of the behavioral experiments in rodents and primates controlling a food dispenser by pushing a lever. Contrary to experiments carried out previously, the long-term tests were realized in the present study. The short-term experiments allow the subject to be concentrated on the task that significantly simplifies data analysis. However, the long-term experiments are required for the real-life applications. Durations of our tests varied from 5 minutes up to 1 hour. Let us stress that in average feeding was taking about 40% of the experimental session in the case of the rats and 35% in the case of the monkeys, the rest of the time was spent by the animals in spontaneous various activities. While the main goal of the study was to discriminate the specific neuronal pattern related to the control action, the additional goals were to make the decision using the single electrode recordings and to predict the control actions rather than detect them. Single electrode classification is desirable for the following self-paced BCI applications with multiple degrees of freedom. Early detection (prediction) allows avoiding the delays of execution.

To achieve the necessary level of selectivity the Multi-way Analysis was chosen since it provides a simultaneous signal processing in several domains. In the present study, we have applied the mapping of the ECoG signal to the temporal-frequency-

spatial space. However, additional modalities, like the phase information, can be easily incorporated. To extract knowledge from the experimental data, a variety of machine learning methods was applied in the BCI research. We have chosen NPLS as a basic approach of multi-modal analysis for extraction the BCI features and for the BCI system calibration. Specially developed for high dimensional and highly correlated observations, this method requires neither exhaustive search of the model nor regularization of the task. It does not depend on the metrics of the BCI system comparison. In our study we have only applied the criterion of the BCI performance to determine the threshold of binarization.

To improve the capacity of the NPLS approach for treatment of high-dimensional data, the ***Iterative NPLS*** algorithm (Eliseyev et al., 2011a; patent received) is introduced in the current study. Having lower memory requirements it allows huge datasets treatment, allows higher resolution of analyses, and preserves the accuracy of the generic algorithm. Moreover, INPLS demonstrates better robustness against noise which can be explained by the overfitting effect suppression. The method was implemented in the BCI system, which demonstrated successful results in series of the close-loop real-time experiments in rats.

Selecting an effective subset of features could significantly optimize efficiency of the model. Proposed ***L1-Penalized NPLS*** algorithm (Eliseyev et al., 2012; patent received) directly includes feature selection in the modeling process. Contrary to other approaches which lead to a linear combination of all features, the L1-PNPLS provides a sparse solution. Applied in the BCI experiments with non-human primates, the method demonstrated efficient selection of the optimal subset of electrodes.

One of the major problems of the BCI studies is significant variability of the neuronal signals, in particular, due to the brain plasticity. These changes in the recorded neuronal activity require recalibration of the BCI systems. The full system recalibration is a time and labor consuming procedure. The adaptive calibration aims to provide a fast adjustment of the BCI system to handle the time changes of the signal. Although the INPLS algorithm allows treating data arrays of huge dimension, this method cannot be applied for the adaptive learning. To overcome the problem, the ***Recursive NPLS*** algorithm (Eliseyev et al., 2011b) is proposed in the study. It allows online processing of the multi-modal data. The method can be efficiency applied for the adaptive learning

to treat the time-dependent recordings. Moreover, the proposed recursive algorithm demonstrated an excellent performance in comparison with NPLS in terms of accuracy (the difference in prediction quality in the real-data experiments did not surpass 0.2%) and convergence rate (in the model experiments the method converged after the first or the second iteration). The requirements for computation resources (memory) are low and do not depend on size of processed data, since the data is treated in the flow. In the real-data experiments, the same predictive quality (less than 0.2% prediction deterioration) was achieved with storing in the active memory only 10 points instead of the whole training set of 1000 points. Thus, RNPLS can be efficiently applied in the BCI systems, which is the perspective of the current study.

The proposed approaches were tested in a set of long-term binary self-paced BCI experiments in animals (rats and non-human primates) and demonstrated promising results for the offline as well as for the open- and close-loop real-time modes (about 80% of correct detections for the rat and up to 70% for the monkey, with acceptably low level of false activation: 1% and 3%, respectively). High frequencies are the most significant for decision making in the cases of both animals. Event-prediction was achieved ~100 ms before the moment of event for the rat, and ~20 ms, for the monkey. Whereas experimental conditions are more complicated than in experiments of other groups, performance characteristics of our methods either surpass or are comparable with others. Detection of event-intention instead of event-detection was achieved. Moreover, only one electrode was used for decision. Computational efficiency of the algorithms allowed their application in the real-time BCI systems. Thus, the suggested approaches provide the essential basis for further development of a human BCI system.

At the same time, additional experiments will allow following studying of the long-term robustness or/and instability of the predictive models, effects of the brain plasticity, etc. Moreover, the experiments in different animals will make it possible to evaluate the discriminative efficiency of the methods for different BCI tasks.

7.2 Perspectives

In the current study a set of methods have been proposed for the binary close-loop real-time BCI. However, the binary BCI represents a particular case of the applications of the suggested approaches. The same methods can be applied for the

multi-class BCI with minimal modifications. The computational efficiency of the proposed algorithms is sufficient enough for the BCI system with multiple degrees of freedom and using a standard portable computer. Moreover, it is possible to apply these methods for the continuous control ($\mathbf{y} \in \mathbb{R}^n$).

The Recursive NPLS algorithm was tested with real data but in the simulation mode. The method showed itself to good advantage and should be applied for the adaptive calibration of the BCI system.

The L1-Penalized NPLS algorithm was used for selection of the effective subset of electrodes. The next step would be the application of this method for other modalities (e.g., frequency).

In the present study, we have applied the INPLS/RNPLS/L1-PNPLS regressions, which are combinations of projection techniques with linear regression. However, the NPLS projections can be coupled with other regression and/or classification techniques that can provide further improvement.

Finally, additional computational optimization should be provided in real-time BCI system. Internal structure of the proposed methods allows an effective code parallelization which will be included in further implementation of our BCI.

List of Publications

Patents

- Aksenova, T., **Yelisyeyev, A.**, 2010. Direct neural interface System and method of calibrating it. Patent PCT/IB2010/001528.
- Aksenova, T., **Yelisyeyev, A.**, 2011. A method calibrating and operating a direct neural interface system. Patent PCT/IB2011/054719.

Publications in peer reviewed journals

- **Eliseyev, A.**, Moro, C., Costecalde, T., Torres, N., Gharbi, S., Mestais, C., Benabid, A.L., Aksenova, T., 2011. Iterative N-way PLS for self-paced BCI in freely moving animals. Journal of Neural Engineering, 8, 046012, 2011.
- **Eliseyev, A.**, Moro, C., Faber, J., Wyss, A., Torres, N., Mestais, C., Benabid, AL., Aksenova, T., 2012. L1-Penalized N-way PLS for subset of electrodes selection in BCI experiments. Submitted to Journal of Neural Engineering.
- Benabid, A.L., Costecalde, T., Torres, N., Moro, C., Aksenova, T., **Eliseyev, A.**, Charvet, G., Sauter, F., Ratel, D., Mestais, C., Pollak, P., Chabardes, S., 2011. Deep brain stimulation BCI at large, where are we going to? Prog Brain Res., 194, 71-82.

Publications in books series

- **Eliseyev, A.**, Benabid, A.L., Aksenova, T., 2011. Recursive Multi-Way PLS for Adaptive Calibration of Brain Computer Interface System. Lecture Notes in Computer Science, 6792/2011, 17-24.

Publications in books of proceedings

- **Eliseyev, A.**, Faber, J., Aksenova, T., 2011. Classification of multi-modal data in a self-paced binary BCI in freely moving animals. EMBC'11, 7147-7150. Boston, 30 August – 3 September, 2011.
- Aksenova, T., **Eliseyev, A.**, Moro, C., Torres, N., Costecalde, T., Charvet, G., Gharbi, S., Mestais, C. Benabid, A.L.; 2010. Tensor based self-paced BCI in freely moving animals. SfN2010, San-Diego, 13-17 November, 2010.
- **Eliseyev, A.**, Moro, C., Faber, J., Wyss, A., Torres, N., Mestais, C., Aksenova, T., Benabid, A.L., 2011. Effective selection of electrode subsets in BCI experiments. IJCCI 2011, Paris, 2011.
- Moro, C., **Eliseyev, A.**, Wyss, A., Faber, J., Torres, N., Costecalde, T., Charvet, G., Sauter, F., Serpollet, A., Mestais, C., Benabid, A.L., Aksenova, T., 2011. ECoG based Self-paced BCI in nonhuman primates performing different motor tasks. SfN 2011, Washington, 2011.

Presentations

- **Eliseyev, A.**, Faber, J., Aksenova, T., 2011. Classification of multi-modal data in a self-paced binary BCI in freely moving animals. EMBC'11, Boston, 30 August – 3 September, 2011.
- **Eliseyev, A.**, Moro, C., Costecalde, T., Torres, N., Gharbi, S., Mestais, C., Benabid, A.L., Aksenova, T. Iterative N-way PLS for real-time control of external effectors with ECoG recordings. Poster at "The Bernstein Conference on Computational Neuroscience 2010".
- **Eliseyev, A.**, Moro, C., Faber, J., Wyss, A., Torres, N., Mestais, C., Aksenova, T., Benabid, A.L., 2011. Effective selection of electrode subsets in BCI experiments. Oral presentation at IJCCI 2011, Paris, 2011.

Acknowledgement

Working toward this thesis during three years I should say it was an interesting and exciting experience. At the same time it would be impossible to finish this study without help, encouragement and support of people around me.

First of all I would like to express my gratitude to Dr. Tatiana Aksenova who was my supervisor and gave me the possibility to do a PhD. Her patience, knowledge and advices had a crucial importance for me.

I would like to thank all the people from the CEA/LETI/CLINATEC team with whom I was working during these years. My special thanks to Prof. Alim-Louis Benabid who is the heart of the project.

In addition, I thank to the CEA/LETI/LE2S laboratory, people from which are productively participating in our project.

Last but not least, my special thanks to my family and all close to me people in Ukraine, France, Russia, Switzerland, and USA who were supporting and encouraging me during the last three years.

The study was supported by Foundation of Nanoscience in frame of the project CE ICoBI.

References

Acar, E., Aykut-Bingol, C., Bingol, H., Bro, R., Yener, B., 2007. Multiway analysis of epilepsy tensors. Bioinformatics, 23(13), i10-i18.

Aizerman, M., Braverman, E., Rozonoer, L., 1964. Theoretical foundations of the potential function method in pattern recognition learning. Automation and remote control, 25, 821-837.

Akaike, H., 1974. A new look at the statistical model identification. IEEE Trans. Automat. Control 19, 716-723.

Albright, R., Cox, J., Duling, D., Langville, A.N., Meyer, C.D., 2006. Algorithms, initializations, and convergence for the nonnegative matrix factorization. Technical report, NCSU Technical report Math 81706.

Anderson, C.W., Sijercic, Z., 1996. Classification of EEG signals from four subjects during five mental tasks Solving Engineering Problems with Neural Networks. Proc. Int. Conf. on Engineering Applications of Neural Networks (EANN'96).

Anderson, C.W., Devulapalli, S.V., Stolz, E.A., 1995. Signal Classification with Different Signal Representations. Neural Networks for Signal Processing, pp. 475-483.

Antoniou, A., 1993. Digital Filters: Analysis, Design, and Application. McGraw-Hill, New-York, NY.

Arroyo, S., Lesser, R.P., Gordon, B., Uematsu, S., Jackson, D., Webber, R., 1993. Functional significance of the mu rhythm of human cortex: an electrophysiologic study with subdural electrodes. Electroencephalography and Clinical Neurophysiology 87 (3), 76–87.

Balakrishnan, D., Puthusserypady, S., 2005. Multilayer perceptrons for the classification of brain computer interface data. Proc. IEEE 31st Annual Northeast Bioengineering Conference.

Barachant, A., Bonnet, S., Congedo, M., Jutten, C., 2011. A Brain-Switch using Riemannian Geometry. 5th International Brain-Computer Interface Conference 2011.

Barker, M., Rayens, W., 2003. Partial least squares for discrimination. Journal of Chemometrics, Volume 17, Issue 3, 166–173.

Barreto, G.A., Frota, R.A., de Medeiros, F.N.S., 2004. On the classification of mental tasks: a performance comparison of neural and statistical approaches. Proc. IEEE Workshop on Machine Learning for Signal Processing.

Barreto, A.B., Taberner, A.M., Vicente, L.M., 1996. Classification of spatio-temporal EEG readiness potentials towards the development of a brain–computer interface. Southeastcon '96. 'Bringing Together Education, Science and Technology', Proc. IEEE.

Bashashati, A., Ward, R.K., Birch, G.E., 2007a. Towards development of a 3-state self-paced brain computer interface. J. Computational Intelligence & Neuroscience, 84386.

Bashashati, A., Fatourechi, M., Ward, R.K., Birch, G.E., 2007b. A survey of signal processing algorithms in brain-computer interfaces based on electrical brain signals. J. of Neural Engineering, 4, R32-R57.

Bashashati, A., Mason, S., Ward, R.K., Birch, G.E., 2006. An improved asynchronous brain interface: making use of the temporal history of the LF-ASD feature vectors. *J. Neural. Eng.* 3**,** 87–94.

Beisteiner, R., Hollinger, P., Lindinger, G., Lang, W., Berthoz, A., 1995. Mental representations of movements. Brain potentials associated with imagination of hand movements. Electroencephal. Clin. Neurophysio., 96, 183–193.

Bennett, K.P., Campbell, C., 2000. Support vector machines: hype or hallelujah? ACM SIGKDD Explor. Newslett., 2, 1–13.

Berger, H., 1929. Über das Elektrenkephalogramm des Menschen. Archiv für Psychiatrie und Nervenkrankheiten 87:527–570.

Biran, R., Martin, D.C., Tresco P.A., 2005. Neuronal cell loss accompanies the brain tissue response to chronically implanted silicon microelectrode arrays. Experimental Neurology 195: 115-26.

Birbaumer, N., 2006. Breaking the silence: brain-computer interfaces (BCI) for communication and motor control. Psychophysiology 43(6), 517-532.

Birbaumer, N., Ghanayim, N., Hinterberger, T., Iversen, I., Kotchoubey, B., Kübler, A., Perelmouter, J., Taub, E., Flor, H., 1999. A spelling device for the paralysed. Nature 398(6725): 297-298.

Birbaumer, N., Elbert, T., Canavan, A.G., Rockstroh, B., 1990. Slow potentials of the cerebral cortex and behavior. Physiological Review 70(1):1–41.

Birch, G.E., 1988. Single trial EEG signal analysis using outlier information. PhD Thesis, The University of British Columbia, Vancouver, Canada.

Bishop, C.M., 1996. Neural Networks for Pattern Recognition. Oxford: Oxford University Press.

Blankertz, B., Sannelli, C., Halder, S., Hammer, E.M., Kübler, A., Müller, K.R., Curio, G., Dickhaus, T., 2010. Neurophysiological predictor of SMR-based BCI performance. NeuroImage 51 (2010) 1303–1309.

Blankertz, B., Dornhege, G., Krauledat, M., Müller, K.R., Kunzmann, V., Losch, F., Curio, G., 2006. The Berlin Brain-Computer Interface: EEG-based communication without subject training. IEEE Transactions on Neural Systems and Rehabilitation Engineering 14(2):147–152.

Blankertz, B., Müller, K.-R., Curio, G., Vaughan, T.M., Schalk, G.,Wolpaw, J.R., Schlögl, A., Neuper, C., Pfurtscheller, G., Hinterberger, T., Schröder, M., Birbaumer, N., 2004. The BCI competition 2003: progress and perspectives in detection and discrimination of EEG single trials. IEEE Trans. Biomed. Eng. 51 (6), 1044–1051.

Blankertz, B., Curio, G., Müller, K.R., 2002. Classifying single trial EEG: towards brain computer interfacing. Adv. Neural Inf. Process. Syst. (NIPS 01), 14, 157–164.

Boostani, R., Moradi, M.H., 2004. A new approach in the BCI research based on fractal dimension as feature and Adaboost as classifier. J. Neural Eng., 1, 212–217.

Borisoff, J.F., Mason, S.G., Bashashati, A., Birch, G.E., 2004. Brain-computer interface design for asynchronous control applications: improvements to the LF-ASD asynchronous brain switch. IEEE Trans. Biomed. Eng., vol. 51, no.6, pp. 985-992.

Bostanov, V., 2004. BCI competition 2003–data sets ib and iib: feature extraction from event-related brain potentials with the continuous wavelet transform and the t-value scalogram. IEEE Trans. Biomed. Eng., 51, 1057–1061.

Boye, A.T., Kristiansen, U.Q., Billinger, M., Nascimento, O.F., Farina, D., 2008. Identification of movement-related cortical potentials with optimized spatial filtering and principal component analysis. Biomedical Signal Processing and Control, 3, 300-304.

Bro, R., 1996. Multiway calibration. Multi-linear PLS. J. Chemom., 10, 47-61.

Bro, R., 1998. Multi-way analysis in the food industry: models, algorithms, and applications. Ph. D. Thesis, Amsterdam, The Netherlands, 1998.

Buch, E., Weber, C., Cohen, L.G., Braun, C., Dimyan, M.A., Ard, T., Mellinger, J., Caria, A., Soekadar, S., Fourkas, A., Birbaumer, N, 2008. Think to Move: a Neuromagnetic Brain-Computer Interface (BCI) System for Chronic Stroke. Stroke 2008; 39; 910-917.

Burges, C.J.C., 1998. A tutorial on support vector machines for pattern recognition. Knowl. Discov. Data Min., 2, 121–167.

Buzsáki, G. 2006. Rhythms of the Brain. Oxford University Press Inc.

Cabrera, A.F., Farina, D., Dremstrup, K., 2010. Comparison of feature selection and classification methods for a brain-computer interface driven by non-motor imagery. Medical and Biological Engineering and Computing, 48, 123-132.

Carpenter, G.A., Grossberg, S., Markuzon, N., Reynolds, J.H., Rosen, D.B., 1992. Fuzzy artmap: a neural network architecture for incrementalsupervised learning of analog multidimensional maps. IEEE Trans. Neural Netw., 3, 698–713.

Chao, Z.C., Nagasaka, Y., Fujii, N., 2010. Long-term asynchronous decoding of arm motion using electrocorticographic signals in monkeys. Frontiers in Neuroengineering, vol. 3, article 3, 1-10.

Chao, Z.C., Nagasaka, Y., Fujii, N., 2009. Long-term asynchronous decoding of 3D hand trajectories using electrocorticographic signals in primates. Proceedings of the 4th International IEEE EMBS Conference on Neural Engineering Antalya, Turkey.

Chapin, J.K., Moxon, K.A., Markowitz, R.S., Nicolelis, M.A., 1999. Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex. Nature America Inc.

Charvet, G., Foerster, M., Filipe, S., Porcherot, J., Bêche, J.F., Guillemaud, R., Audebert, P., Régis, G., Zongo, B., Robinet, S., Condemine, C., Têtu, Y., 2011. WIMAGINE: A Wireless, Low Power, 64-Channel ECoG Recording Platform for

Implantable BCI Applications. 5th International IEEE EMBS Conference on Neural Engineering.

Cheng, M., Gao, X., Gao, S., Xu, D., 2002. Design and implementation of a brain–computer interface with high transfer rates. IEEE Trans. Biomed. Eng. 49, 1181–1186.

Chiappa, S., Bengio, S., 2004. HMM and IOHMM modeling of EEG rhythms for asynchronous BCI systems. European Symposium on Artificial Neural Networks ESANN.

Cichocki, A., Zdunek, R., Phan, A.H., Amari, S., 2009. Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation. John Wiley & Sons, Ltd, 2009.

Cincotti, F., Scipione, A., Tiniperi, A., Mattia, D., Marciani, M.G., Millán, J.R., Salinari, S., Bianchi, L., Babiloni, F., 2003. Comparison of different feature classifiers for brain computer interfaces. Proc. 1st Int. IEEE EMBS Conf. on Neural Engineering.

Comon, P., Berge, J., 2006. Generic and Typical Ranks of Three-Way Arrays. I3S report ISRN I3S/RR-2006-29-FR.

Comon, P., 2002. Tensor Decompositions, State of the Art and Applications. Mathematics in Signal Processing V, J. G. McWhirter and I. K. Proudler (Ed.) (2002) 1-24.

Congedo, M., Lotte, F., Lécuyer, A., 2006. Classification of movement intention by spatially filtered electromagnetic inverse solutions. Phys. Med. Biol., 51, 1971–1989.

Cook, R.D., Weisberg, S., 1982. Residuals and Influence in Regression. London: Chapman and Hall.

Cortes, C., Vapnik, V., 1995. Support-Vector Networks. Machine Learning, 20.

Coyle, S.M., Ward, T.E., Markham, C.M., 2007. Brain–computer interface using a simplified functional near-infrared spectroscopy system. J. Neural Eng. 4, 219–226.

Coyle, S.M., Ward, T.E., Markham, C.M., McDarby, G., 2004. On the suitability of near-infrared (NIR) systems for next-generation brain-computer interfaces. Physiological Measurement 25(4):815– 822.

Dayal, B.S., MacGregor, J.F., 1997. Improved PLS algorithm. J. Chemometrics, 11, 73—85.

De Jong, S., 1998. Regression coefficients in multilinear PLS. J. Chemom., 12, 77–81.

Devijver, P.A., Kittler, J., 1982. Pattern Recognition: A Statistical Approach. Prentice-Hall, London, 1982.

Dijon, J., Boronat, J.F., Fournier, A., De Monsabert, T.G., Montmayeul, B., Levis, M., Levy, F., Sarrasin, D., Meyer, R., Dean, K.A., Coll, B.F., Johnson, S.V., Hagen, C., Jaskie, J.E., 2004. Towards A Low Cost High Quality Carbon Nanotubes Field Emission Display. SID 04 DIGEST, 820–823.

Dornhege, G., Krauledat, M., Müller, K.R., Blankertz, B., 2007. General Signal Processing and Machine Learning Tools for BCI Analysis. Toward Brain-Computer Interfacing. MIT Press, Cambridge, MA, pp. 207-234.

Duda, R.O., Hart, P.E., Stork, D.G., 2001. Pattern Recognition. 2nd edn, New York: Wiley-Interscience.

Ejserholm, F., Köhler, P., Bengtsson, M., Jorntell, H., Schouenborg, J., Wallman, L., 2011. A Polymer Based Electrode Array for Recordings in the Cerebellum. 5th International IEEE EMBS Conference on Neural Engineering.

Eliseyev, A., Moro, C., Faber, J., Wyss, A., Torres, N., Mestais, C., Benabid, AL., Aksenova T., 2012. L1-Penalized N-way PLS for subset of electrodes selection in BCI experiments. Submitted to Journal of Neural Engineering.

Eliseyev, A., Moro, C., Costecalde, T., Torres, N., Gharbi, S., Mestais, C., Benabid, A.L., Aksenova, T., 2011a. Iterative N-way PLS for self-paced BCI in freely moving animals. Journal of Neural Engineering, 8, 046012.

Eliseyev, A., Benabid, A.L., Aksenova, T., 2011b. Recursive Multi-Way PLS for Adaptive Calibration of Brain Computer Interface System. Lecture Notes in Computer Science, 6792/2011, 17-24.

Essl, M., Rappelsberger, P., 1998. EEG coherence and reference signals: experimental results and mathematical explanations. Med. Biol. Eng. Comput., vol. 36, no.4, pp. 399-406.

Fan, J., 1997. Comments on "Wavelet in Statistics: A Review." by Antoniadis. J. Italian Statist. Assoc. 6, 131-138.

Farwell, L.A., Donchin, E., 1988. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. Electroencephalography and Clinical Neurophysiology 70(6):510–523.

Fatourechi, M., Ward, R.K., Birch, G.E., 2008. A self-paced brain–computer interface system with a low false positive rate. J Neural Eng 5:9–23. doi:10.1088/1741-2560/5/1/002.

Fatourechi, M., Birch, G.E., Ward, R.K., 2007. A Self-paced Brain Interface System that Uses Movement Related Potentials and Changes in the Power of Brain Rhythms. Journal of Computational Neuroscience, Vol.23, No.1, pp.21-37.

Fatourechi, M., Bashashati, A., Birch, G.E., Ward, R.K., 2006. Automatic user customization for improving the performance of a self-paced brain interface system, Med Bio Eng Comput 44, 1093–1104.

Felton E.A., Wilson J.A., Williams J.C., Garell P.C., 2007. Electrocorticographically controlled brain–computer interfaces using motor and sensory imagery in patients with temporary subdural electrode implants. Report of four cases. J. Neurosurg. 106 495–500.

Felzer, T., Freisieben, B., 2003. Analyzing EEG signals using the probability estimating guarded neural classifier. IEEE Trans. Neural Syst. Rehabil. Eng., 11, 361–371.

Friedman, J.H.K., 1997. On bias, variance, 0/1-loss, and the curse-of-dimensionality. Data Min. Knowl. Discov. 1 55–77.

Fukunaga, K., 1990. Statistical Pattern Recognition. 2nd edn, New York: Academic.

Galán, F., Nuttin, M., Vanhooydonck, D., Lew, E., Ferrez, P.W., Philips, J., Millán, J. del R., 2008. Continuous Brain-Actuated Control of an Intelligent Wheelchair by Human EEG. In proceedings, 4th Intl. Brain-Computer Interface Workshop and Training Course, 2008.

Gao, Y., Black, M.J., Bienenstock, E., Wu, W., Donoghues, J.P., 2003. A Quantitative Comparison of Linear and Non-linear Models of Motor Cortical Activity for the Encoding and Decoding of Arm Motions. Proceedings of the 1st International IEEE EMBS Conference on Neural Engineering.

Garcia, G., Ebrahimi, T., Vesin, J.M., 2003a. Correlative exploration of EEG signals for direct brain-computer communications. Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (Hong Kong), pp. 816-819.

Garcia, G., Ebrahimi, T., Vesin, J.M., 2003b. Support vector EEG classification in the fourier and time-frequency correlation domains. Proc. 1st IEEE-EMBS Conf. on Neural Engineering (Capri Island, Italy), pp. 591-594.

Garrett, D., Peterson, D.A., Anderson, C.W., Thaut, M.H., 2003. Comparison of linear, nonlinear, and feature selection methods for EEG signal classification. IEEE Trans. Neural Syst. Rehabil. Eng., 11, 141–144.

Geladi, P., Kowalski, B.R., 1986. Partial least-squares regression: a tutorial. Analytica Chimica Acta, 185, 1-17.

Givens, W., 1958. Computation of plane unitary rotations transforming a general matrix to triangular form. J. SIAM 6(1) , pp. 26–50.

Golub, G.H., Van Loan, C.F., 1996. Matrix Computations (3rd ed.), Johns Hopkins, ISBN 978-0-8018-5414-9.

Golub, G., Heath, M., Wahba, G., 1979. Generalization cross-validation as a method for choosing a good ridge parameter. Technometrics 21, 215-223.

Graimann, B., Huggins, J., Levine, S., Pfurtscheller, G., 2004. Toward a direct brain interface based on human subdural recordings and wavelet-packet analysis. IEEE Transactions on Biomedical Engineering 51(6):954–962.

Gu, Y., Farina, D., Murguialday, A.R., Dremstrup, K., Montoya, P., Birbaumer, N., 2009. Offline identification of imaged speed of wrist movements in paralyzed ALS patients from single-trial EEG. Frontiers in Neuroscience, vol. 3, article 62, 1-7.

Hamedani, G.G., Tata, M.N., 1975. On the determination of the bivariate normal distribution from distributions of linear combinations of the variables. The American Mathematical Monthly, 82, 913–915.

Harshman, R.A., 1970. Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis. UCLA Working Papers in Phonetics, 16, 84.

Haselsteiner, E., Pfurtscheller, G., 2000. Using time-dependant neural networks for EEG classification. IEEE Trans. Rehabil. Eng., 8, 457–463.

Herrmann, C.S., 2001. Human EEGresponses to 1-100 Hz flicker: Resonance phenomena in visual cortex and their potential correlation to cognitive phenomena. Experimental Brain Research V137(3):346–353.

Hilbe, J.M., 2009. Logistic Regression Models. Chapman & Hall/CRC Press.

Hill, N.J., Lal, T.N., Schröder, M., Hinterberger, T., Wilhelm, B., Nijboer, F., Mochty, U., Widman, G., Elger, C., Schölkopf, B., Kübler, A., Birbaumer, N., 2006. Classifying EEGand ECoG signals without subject training for fast BCI

implementation: Comparison of nonparalyzed and completely paralyzed subjects. IEEE Transactions on Neural Systems and Rehabilitation Engineering 14(2):183–186.

Hinterberger, T., Kübler, A., Kaiser, J., Neumann, N., Birbaumer, N, 2003. A brain-computer interface (BCI) for the locked-in: comparison of different EEG classifications for the thought translation device. Clin Neurophysiol., 114(3):416-25.

Hiraiwa, A., Shimohara, K., Tokunaga, Y., 1990. EEG topography recognition by neural networks. IEEE Eng. Med. Biol. Mag., 9, 39–42.

Hirata, M., Yanagisawa, T., Goto, T., Matsushita, K., Suzuki, T., Yokoi, H., Sakura, O., Yoshida, T., Sato, F., Sawada, J., Saitoh, Y., Kishima, H., Ushiba, J., Kamitani, Y., Yoshimine, T., 2010. An integrative BMI approach for functional restoration using human electrocorticograms. SfN2010.

Hochberg, L.R., Serruya, M.D., Friehs, G.M., Mukand, J.A., Saleh, M., Caplan, A.H., Branner, A., Chen, D., Penn, R.D., Donoghue, J.P., 2006. Neuronal ensemble control of prosthetic devices by a human with tetraplegia. Nature 442(7099):164–171.

Hoerl, A.E., Kennard, R.W., 2000. Ridge Regression: biased estimation for nonorthogonal problem. Technometrics, 42, 80-86.

Hoffmann, U., 2007. Bayesian machine learning applied in a brain-computer interface for disabled users. Ph. D. Thesis, Lausanne, Switzerland, 2007.

Hoffmann, U., Garcia, G., Vesin, J.M., Diserens, K., Ebrahimi, T., 2005. A boosting approach to p300 detection with application to brain–computer interfaces. Conference Proc. 2nd Int. IEEE EMBS Conf. Neural Engineering.

Householder, A.S., 1958. Unitary Triangularization of a Nonsymmetric Matrix. Journal of the ACM 5 (4): 339–342.

Hoya, T., Hori, G., Bakardjian, H., Nishimura, T., Suzuki, T., Miyawaki, Y., Funase, A., Cao, J., 2003. Classification of single trial EEG signals by a combined principal + independent component analysis and probabilistic neural network approach. Proc. ICA2003, pp. 197–202.

Huggins, J.E., Graimann, B., Chun, S.Y., Fessler, J,A., Levine, S.P., 2007. Electrocorticogram as a Brain Computer Interface Signal Source. In Towards Brain-Computer Interfacing, edited by G. Dornhege, J.dR. Millan, T. Hinterberger, D. McFarland, and K.R. Meuller, MIT Press, pp. 129-145, 2007.

Huggins, J., Levine, S.P., BeMent, S.L., Kushwaha, R.K., Schuh, L.A., Passaro, E.A., Rohde, M.M., Ross, D.A., Elisevich, K.V., Smith, B.J., 1999. Detection of Event-Related Potentials for Development of a Direct Brain Interface. Journal of Clinical Neurophysiology: 16(5), 448.

Jain, A.K., Duin, R.P.W., Mao, J., 2000. Statistical pattern recognition: a review. IEEE Trans. Pattern Anal. Mach. Intell., 22, 4–37.

Kaper, M., Meinicke, P., Grossekathoefer, U., Lingner, T., Ritter, H., 2004. BCI competition 2003–data set iib: support vector machines for the p300 speller paradigm. IEEE Trans. Biomed. Eng., 51, 1073–1076.

Kauhanen, L., Nykopp, T., Lehtonen, J., Jylänki, P., Heikkonen, J., Rantanen, P., Alaranta, H., Sams, M., 2006. EEG and MEG brain-computer interface for tetraplegic patients. IEEE Transactions on Neural Systems and Rehabilitation Engineering 14(2):190–193.

Kayser, J., Fong, R., Tenke, C.E., Bruder, G.E., 2003. Event-related brain potentials during auditory and visual word recognition memory tasks. Brain Res Cogn Brain Res, 16, 11-25.

Keefer, E.W., Botterman, B.R., Romero, M.I., Rossi, A.F., Gross, G.W., 2008. Carbon nanotubes coating improves neuronal recordings. Nature nanotechnology, 3.

Keirn, Z.A., Aunon, J.I., 1990. A new mode of communication between man and his surroundings. IEEE Trans. Biomed. Eng., 37, 1209–1214.

Kellis, S., Miller, K., Thomson, K., Brown, R., House, P., Greger, B., 2010. Decoding spoken words using local field potentials recorded from the cortical surface. Journal of Neural Engineering, 7(5), 1-10.

Kohonen, T., 1990. The self-organizing maps. Proc. IEEE, 78, 1464–1480.

Kolda, T.G., Bader, B.W., 2007. Tensor Decompositions and applications. Sandia report, SAND2007-6702.

Kostov, A., Polak, M., 2000. Parallel man-machine training in development of EEG-based cursor control. IEEE Trans. Rehabil. Eng., 8, 203–205.

Kronegg, J., Voloshynovskiy, S., Pun, T., 2005. Analysis of bit-rate definitions for Brain-Computer Interfaces. CSREA HCI 2005: 40-46.

Kubánek, J., Miller, K.J., Ojemann, J.G., Wolpaw, J.R., Schalk, G., 2009. Decoding flexion of individual fingers using electrocorticographic signals in humans. Journal of Neural Engineering, 6(6), 1-14.

Kübler, A., Müller, K.R., 2007. An introduction to brain computer interfacing. In: Dornhege, G., del, R., Millán, J., Hinterberger, T., McFarland, D., Müller, K.R. (eds.) Toward Brain-Computer Interfacing, pp. 1-25. MIT press, Cambridge (2007).

Kübler, A., Nijboer, F., Mellinger, J., Vaughan, T.M., Pawelzik, H., Schalk G., McFarland, D.J., Birbaumer, N., Wolpaw, J.R., 2005. Patients with ALS can use sensorimotor rhythms to operate a brain-computer interface. Neurology, 64(10):1775-1777.

Kübler, A., Kotchoubey, B., Hinterberger, T., Ghanayim, N., Perelmouter, J., Schauer, M., Fritsch, C., Taub, E., Birbaumer, N., 1999. The thought translation device: a neurophysiological approach to communication in total motor paralysis. Exp Brain Res 124(2): 223-232.

Kutsy, R.L., 1999. TI Focal extratemporal epilepsy: clinical features, EEG patterns, and surgical approach. Journal of the Neurological Sciences 166:1-15.

Lal, T.N., Schröder, M., Hill, J., Preissl, H., Hinterberger, T., Mellinger, J., Bogdan, M., Rosenstiel, W., Hofmann, T., Birbaumer, N., Schölkopf, B., 2005. A brain computer interface with online feedback based on magnetoencephalography. In Proceedings of the 22nd International Conference on Machine Learning: 465–472.

Lalor, E., Kelly, S.P., Finucane, C., Burke, R., Smith, R., Reilly, R.B., McDarby G., 2005. Steady-state VEP-based brain-computer interface control in an immersive 3D gaming environment. EURASIP Journal on Applied Signal Processing 2005(19):3156–3164.

Land, S., Friedman, J., 1996. Variable fusion: a new method of adaptive signal regression. Technical Report. Department of Statistics, Stanford University, Stanford.

Lebedev, M.A., Nicolelis, M.A.L., 2006. Brain-machine interfaces: Past, present and future. Trends in Neurosciences 29(9):536–546.

Lee, H., Choi, S., 2003. PCA+HMM+SVM for EEG pattern classification. Proc. 7th Int. Symp. on Signal Processing and its Applications.

Leeb, R., Settgast, V., Fellner, D.W., Pfurtscheller, G., 2007. Self-paced exploring of the Austrian National Library through thoughts. International Journal of Bioelectromagnetism, 9, 237-244.

Lemm, S, Schafer, C., Curio, G., 2004. BCI competition 2003–data set iii: probabilistic modeling of sensorimotor mu rhythms for classification of imaginary hand movements. IEEE Trans. Biomed. Eng., 51, 1077–1080.

Leuthardt, E.C., Miller, K.J., Schalk, G., Rao, R.P., Ojemann, J.G., 2006. Electrocorticography-based brain computer interface--the Seattle experience. IEEE Trans Neural Syst Rehabil Eng 14(2): 194-198.

Leuthardt E.C., Schalk G., Wolpaw J.R., Ojemann J.G., Moran D.W., 2004. A brain-computer interface using electrocorticographic signals in humans. J Neural Eng. 1(2): 63-71.

Levine, S.P., Huggins, J.E., BeMent, S.L., Kushwaha, R.K., Schuh, L.A., Passaro, E.A., Rohde, M.M., Ross, D.A., 1999. Identification of Electrocorticogram Patterns as the Basis for a Direct Brain Interface. Journal of Clinical Neurophysiology: 16(5), 439-447.

Li, J. and Zhang, L., 2010. Regularized tensor discriminant analysis for single trial EEG classification in BCI Pattern Recognition Letters, 2010, 31, 619-628.

Li, J., Zhang, L., Tao, D., Sun, H., Zhao, Q., 2009. A prior neurophysiologic knowledge free tensor-based scheme for single trial EEG classification. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 17, 107-115.

Lotte, F., Congedo, M., Lécuyer, A., Lamarche, F., Arnaldi, B., 2007. A review of classification algorithms for eeg-based brain-computer interfaces. J. Neural Eng., vol. 4, pp. R1-R13.

Ludwig, K.A., Miriani, R.M., Langhals, N.B., Joseph, M.D., Anderson, D.J., Kipke, D.R., 2009. Using a Common Average Reference to Improve Cortical Neuron Recordings From Microelectrode Arrays. J Neurophysiol, 101, 1679-1689.

Makeig, S., Westerfield, M., Townsend, J., Jung, T.P., Courchesne, E., Sejnowski, T.J., 1999. Functionally independent components of early event-related potentials in a visual spatial attention task. Philos Trans R Soc Lond B Biol Sci, 354, 1135-1144.

Margalit, E., Weiland, J.D., Clatterbuck, R.E., Fujii, G.Y., Maia, M., Tameesh, M., Torres, G., D'Anna, S.A., Piyathaisere, D.V., Desai, S., Olivi, A., de Juan, E.Jr.,

Humayun, M.S., 2003. Visual and electrical evoked response recorded from subdural electrodes implanted above the visual cortex in normal dogs under two methods of anesthesia. Journal of Neuroscience Methods, 123(2), 129-137.

Martens, H., Næs, T., 1989. Maltivariate Calibration. John Wiley & Sons: New York, 1989.

Martin, J., 1991. The collective electrical behavior of cortical neurons: The electroencephalogram and the mechanisms of epilepsy. Principles of neural science. Elsevier, 777-790.

Martínez-Montes, E., Sánchez-Bornot, J.M., Valdés-Sosa, P.A., 2008. Penalized PARAFAC analysis of spontaneous EEEG recordings. Statistica Sinica, 18, 1449-1464.

Mason, S.G., Birch, G.E., 2005. Temporal control paradigms for direct brain interfaces – rethinking the definition of asynchronous and synchronous. In Proceedings of HCI International, Las Vegas.

Mason, S.G., Birch, G.E., 2000. A brain-controlled switch for asynchronous control applications. IEEE Trans. Biomed.Eng. 47, 1297–1307.

McFarland, D.J., Krusienski, D.J., Wolpaw, J.R., 2006. Brain-computer interface signal processing at the Wadsworth Center: mu and sensorimotor beta rhythms. Progress in Brain Research, 159, 411-419.

McFarland, D.J., Miner, L.A., Vaughan, T.M., Wolpaw, J.R., 2000. Mu and beta rhythm topographies during motor imagery and actual movements. Brain Topogr., 12, 177.

McFarland, D.J., McCane, L.M., David, S.V., Wolpaw, J.R., 1997. Spatial filter selection for EEG-based communication. Electroencephalogr. Clin. Neurophysiol., vol. 103, no.3, pp. 386-394.

Millán, J.R., Renkens, F, Mourino, J., Gerstner, W., 2004. Noninvasive brain-actuated control of a mobile robot by human. EEG IEEE Trans. Biomed. Eng., 51, 1026–1033.

Millán, J.R., Mouriño, J., Cincotti, F., Babiloni, F., Varsta, M., Heikkonen, J., 2000. Local neural classifier for EEG-basedrecognition of mental tasks. IEEE-INNS-ENNS Int. Joint Conf. on Neural Networks.

Miller, K., Leuthardt, E., Schalk, G., Rao, R., Anderson, N., Moran, D., Miller, J., Ojemann, J., 2007. Spectral Changes in Cortical Surface Potentials during Motor Movement. Journal of Neuroscience 27:2424-2432.

Mørup, M., Hansen, L.K., Arnfred, S.M., 2008. Algorithms for Sparse Nonnegative Tucker Decomposition. Neural Computation, 20, 2112-2131.

Müller, K.R., Krauledat, M., Dornhege, G., Curio, G., Blankertz, B., 2004. Machine learning techniques for brain–computer interfaces. Biomed. Technol., 49, 11–22.

Müller-Putz, G.R., Kaiser, V., Solis-Escalante, T., Pfurtscheller, G., 2010. Fast set-up asynchronous brain-switch based on detection of foot motor imagery in 1-channel EEG. International Federation for Medical and Biological Engineering 2010, 10.1007/s11517-009-0572-7.

Nascimento, O.F., Dremstrup, K., Voigt, M., 2006. Movement-related parameters modulate cortical activity during imaginary isometric plantar-flexions. Experimental Brain Research, 171, 78-90.

Nazarpour, K., 2008. Brain Signal Analysis in Space-Time-Frequency Domain; An Application to Brain Computer Interfacing. PhD Thesis, Cardiff University, UK.

Nazarpour, K., Sanei, S., Shoker, L., Chambers, J.A., 2006. Parallel space-time-frequency decomposition of eeg signals for brain computer interfacing. EUSIPCO 2006.

Nicolelis, M.A.L., Dimitrov, D., Carmena, J.M., Crist, R., Lehew, G., Kralik, J.D., Wise, S.P., 2003. Chronic, multisite, multielectrode recordings in macaque monkeys. Proceedings of the National Academy of Sciences (PNAS) 100(19):11041–11046.

Nieuwenhuis, S., Aston-Jones, G., Cohen, J.D., 2005. Decision making, the P3, and the locus coeruleus-norepinephrine system. Psychological Bulletin, 131, 510–532.

Nilsson, J., de Jong, S., Smilde, A.K., 1997. Multiway calibration in 3D QSAR. Journal of Chemometrics, 11(6), 511-524.

Normann, R.A., Maynard, E.M., Rousche, P.J., Warren, D.J., 1999. A neural interface for a cortical vision prosthesis. Vision Res 39: 2577-2587.

Obermeier, B., Guger, C., Neuper, C., Pfurtscheller, G., 2001. Hidden Markov models for online classification of single trial EEG. Pattern Recognit. Lett., 1299–1309.

Obermaier, B., Neuper, C., Guger, C., Pfurtscheller, G., 2000. Information transfer rate in a five-classes brain–computer interface. IEEE Trans. Neural Syst. Rehabil. Eng., 9, 283–288.

Oppenheim, A.V., Schafer, R.W., 1989. Discrete-time signal processing. Prentice HallSignal Processing Series. Prentice Hall.

Palaniappan, R., 2005. Brain computer interface design using band powers extracted during mental tasks. Proc. 2nd Int. IEEE EMBS Conf. on Neural Engineering.

Palaniappan, R., Paramesran, R., Nishida, S., Saiwaki, N., 2002. A new brain–computer interface design using fuzzy artmap. IEEE Trans. Neural Syst. Rehabil. Eng., 10, 140–148.

Penny, W.D., Roberts, S.J., Curran, E.A., Stokes, M.J., 2000. EEG-based communication: a pattern recognition approach. IEEE Trans. Rehabil. Eng., 8, 214–215.

Pfurtscheller, G., Allison, B.Z., Brunner, C., Bauernfeind, G., Solis-Escalante, T., Scherer, R., Zander, T.O., Mueller-Putz, G., Neuper, C., Birbaumer, N., 2010a. The hybrid BCI. Frontiers in Neuroscience, 4(42), 1-11.

Pfurtscheller, G., Solis-Escalante, T., Ortner, R., Linortner, P., Muller-Putz, G.R., 2010b. Self-Paced Operation of an SSVEP-Based Orthosis With and Without an Imagery-Based "Brain Switch:" A Feasibility Study Towards a Hybrid BCI. Neural Systems and Rehabilitation Engineering, 18(4), 409-414.

Pfurtscheller, G., Solis-Escalante, T., 2009. Could the beta rebound in the EEG be suitable to realize a "brain switch"? Clin. Neurophysiol. 120, 24–29.

Pfurtscheller, G., Neuper, C., Brunner, C., Lopes da Silva, F.H., 2005. Beta rebound after different types of motor imagery in man. Neurosci. Lett. 378, 156–159.

Pfurtscheller, G, Graimann, B, Huggins, J.E., Levine, S.P., 2004. Brain-computer communication based on the dynamics of brain oscillations. Suppl Clin Neurophysiol., 57, 583-591.

Pfurtscheller, G., Neuper, C., 2001. Motor imagery and direct brain-computer communication. Proceedings of the IEEE 89(7):1123–1134.

Pfurtscheller, G., 1999. EEG event-related desynchronization (ERD) and event-related synchronization (ERS) Electroencephalography: Basic Principles, Clinical

Applications and Related Fields. 4th edn, ed E. Niedermeyer and F. H. Lopes da Silva (Baltimore, MD: Williams and Wilkins), pp. 958–967.

Pfurtscheller, G., da Silva, F.H.L., 1999. Event-Related Desynchronisation, Handbook of Electroencephalogr. Clin. Neurophysiol. Revised Series, Elsevier, Amsterdam.

Pfurtscheller, G., Flotzinger, D., Kalcher, J., 1993. Brain–computer interface-a new communication device for handicapped persons. J. Microcomput. Appl., 16, 293–299.

Pine J., 2006. A History of MEA Development. In: Baudry M, Taketani M, eds. Advances in Network Electrophysiology Using Multi-Electrode Arrays. New York: Springer Press, 3-23.

Pistohl, T., Ball, T., Schulze-Bonhage, A., Aertse, A., Mehring, C., 2008. Prediction of arm movement trajectories from ECoG-recordings in humans. Journal of Neuroscience Methods, 167(2008), 105-114.

Qian, K., Nikolov, P., Huang, D., Fei, D.Y., Chen, X., Bai, O, 2010. A motor imagery-based online interactive brain-controlled switch: paradigm development and preliminary test. Clinical Neurophysiology, 121, 1303-1313.

Qin, J., Li, Y., Cichocki, A., 2005. ICA and committee machine-based algorithm for cursor control in a BCI system. Lecture Notes in Computer Science.

Qin, S.J., 1998. Recursive PLS algorithms for adaptive data modeling. Computers chem. Engng, 22, 503—514.

Rabiner, L.R., 1989. A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE, 77, 257–286.

Rakotomamonjy, A., Guigue, V., Mallet, G., Alvarado, V., 2005. Ensemble of SVMs for improving brain computer interface P300 speller performances. In International Conference on Artificial Neural Networks.

Ramoser, H., Muller-Gerking, J., Pfurtscheller, G., 2000. Optimal spatial filtering of single trial EEG during imagined hand movement. IEEE Trans. Rehabil. Eng., vol. 8, no.4, pp. 441-446.

Ramsey, N., Van De Heuvel, M., Kho, K., Leijten, F., 2006. Towards human BCI applications based on cognitive brain systems: An investigation of neural signals recorded from the dorsolateral prefrontal cortex. IEEE Transactions on Neural Systems and Rehabilitation Engineering 14(2):214–217.

Rijsbergen, C.J., 1979. Information retrieval. Available at http://www.dcs.gla.ac.uk/˜iain/keith/.

Rouse, A.G. and Moran, D.W., 2009. Neural adaptation of epidural electrocorticographic (EECoG) signals during closed-loop brain computer interface (BCI) tasks. Conf Proc IEEE Eng Med Biol Soc 2009: 5514-5517.

Salanova, V., Morris, H.H., Van Ness, P.C., Luders, H., Dinner, D., Wyllie, E., 1993. Comparison of scalp electroencephalogram with subdural electrocorticogram recordings and functional mapping in frontal lobe epilepsy. Archives of Neurology, 50(3), 294-299.

Sanei, S., Chambers, J.A., 2007. EEG Signal Processing. Wiley, New York.

Sauter-Starace, F., Bibari, O., Berger, F., Caillat, P., Benabid, A.L., 2009. ECoG recordings of a non-human primate using carbon nanotubes electrodes on a flexible polyimide implant. Neural Engineering, 2009. NER '09.

Sauter-Starace, F., Bibari, O., Ram, N., Chabrol, C., Charvet, G., Marchand, G., Pudda, C., Liu, J., Benabid, A.L., Divoux, J.L., Roberton, R., Pisella, C., 2006. Projet NEUROLINK: vers une interface cerveau machine.

Schalk, G., Miller, K.J., Anderson, N.R., Wilson, J.A., Smyth, M.D., Ojemann, J.G., Moran, D.W., Wolpaw, J.R., Leuthardt, E.C., 2008. Twodimensional movement control using electrocorticographic signals in humans. Journal of Neural Engineering, 5, 75-84.

Scherer, R., Lee, F., Schlgl, A., Leeb, R., Bischof, H., Pfurtscheller, G., 2008. Toward self-paced brain-computer communication: navigation through virtual worlds. IEEE Transactions on Biomedical Engineering, 55, 675-682.

Scherer, R., Schloegl, A., Lee, F., Bischof, H., Janša, J., Pfurtscheller, G., 2007. The Self-Paced Graz Brain-Computer Interface: Methods and Applications. Comput Intell Neurosci. 2007; 2007: 79826.

Scherer, R., Müller, G.R., Neuper, C., Graimann, B., Pfurtscheller, G. 2004. An asynchronously controlled EEG-based virtual keyboard: improvement of the spelling rate. IEEE Trans Biomed Eng., 51, 979-984.

Scherer, R., Graimann, B., Huggins, J.E., Levine, S.P., Pfurtscheller G., 2003. Frequency component selection for an ECoG-based brain-computer interface. Biomed Tech (Berl) 48(1-2): 31-36.

Schlögl, A., Kronegg, J., Huggins, J., Mason, S.G., 2007. Evaluation criteria in BCI research. In Towards Brain-Computer Interfacing (G. Dornhege, J. R. Millan, T. Hinterberger, D. McFarland and K. R. Muller, Eds.), MIT Press, 2007.

Schlögl, A., Lee, F., Bischof, H., Pfurtscheller, G., 2005. Characterization of four-class motor imagery EEG data for the BCI-competition 2005. Journal of Neural Engineering 2(4): L14-L22.

Schmidt, M., 2005. Least Squares Optimization with L1-Norm Regularization. Cs542B Project Report, December 2005.

Schuh, L., Drury, M., 1996. Intraoperative Electrocorticography and Direct Cortical Electrical Stimulation. Seminars in Anesthesia, 16, 46-55.

Schwartz, A.B., 2004. Cortical neural prosthetics. Annual Review of Neuroscience 27: 487–507.

Schwartz, G., 1978. Estimating the dimension of a model. Ann. Statist. 6, 461-464.

Serruya, M.D., Hatsopoulos, N.G., Paninski, L., Fellows, M.R., Donoghue, J.P., 2002. Instant neural control of a movement signal. Nature 416(6877): 141-142.

Shenoi, B.A., 2006. Introduction to digital signal processing and filter design. John Wiley and Sons.

Sherwood, J., Derakhshani, R., 2009. On Classifiability of Wavelet Features for EEG-Based Brain-Computer Interfaces. Proceedings of the 2009 international joint conference on Neural Networks, 2508-2515.

Shevade, S.K., Keerthi, S.S., 2003. A simple and efficient algorithm for gene selection using sparse logistic regression. Bioinformatics, 19(17), 2246-2253.

Sitaram, R., Zhang, H., Guan, C., Thulasidas, M., Hoshi, Y., Ishikawa, A., Shimizu, K., Birbaumer, N., 2007. Temporal classification of multichannel near-infrared spectroscopy signals of motor imagery for developing a brain-computer interface. NeuroImage 34(4): 1416–1427.

Sitaram, R., Hoshi, Y., Guan, C., 2005. Near infrared spectroscopy based brain-computer interfaces, vol. 5852. Bellingham, Wash.: Society of Photo-Optical Instrumentation Engineers.

Solhjoo, S., Nasrabadi, A.M., Golpayegani, M.R.H., 2005. Classification of chaotic signals using HMM classifiers: EEG-based mental task classification. Proc. European Signal Processing Conference.

Solhjoo, S., Moradi, M.H., 2004. Mental task recognition: a comparison between some of classification methods. BIOSIGNAL 2004 Int. EURASIP Conference.

Solis-Escalante, T., Müller-Putz, G.R., Brunner, C., Kaiser, V., Pfurtscheller, G., 2010. Analysis of sensorimotor rhythms for the implementation of a brain switch for healthy subjects. Biomed. Signal Process. Control 5, 15–20.

Squires, K.C., Wickens, C., Squires, N.K., Donchin, E., 1976. The effect of stimulus sequence on the waveform of the cortical event-related potential. Science 193(4258): 1142–1146.

Srinivasan, R., 1999. Methods to improve the spatial resolution of EEG. International Journal of Bioelectromagnetism, 1, 102-111.

Sutter, E.E., 1992. The brain response interface: communication through visually-induced electrical brain responses. Journal of Microcomputer Applications 15(1), 31-45.

Tangermann, M., 2008. Contributions from mathematics: Applying machine learning algorithms to BCI. Brain Computer Interfacing in 2008, Utrecht The Netherlands.

Tangermann, M., 2007. Feature Selection for Brain-Computer Interfaces. PhD Thesis, Eberhard-Karls-Universität, Tübingen, Germany.

Taylor, D.M., 2007. The importance of online error correction and feed-forward adjustment in brain-machine interfaces for restoration of movement. In Towards Brain-Computer Interfacing (G. Dornhege, J. R. Millan, T. Hinterberger, D. McFarland and K. R. Muller, Eds.), MIT Press, 2007.

Taylor, D., Tillery, S., Schwartz, A., 2002. Direct cortical control of 3D neuroprosthetic devices. Science 296(5574):1829–1832.

Teolis, A., 1998. Computational Signal Processing with Wavelets. Birkhauser.

Tibshirani, R., 1996. Regression shrinkage and variable selection via the lasso. J. Roy. Statist. Soc. Ser. B 58, 267-288.

Torres Valderrama, A., Oostenveld, R., Vansteensel, M.J., Huiskamp, G.M., Ramsey, N.F., 2010. Gain of the human dura in vivo and its effects on invasive brain signal feature detection. J Neurosci Methods 187(2): 270-279.

Townsend, G., Graimann, B., Pfurtscheller, G., 2004. Continuous EEG classification during motor imagery-simulation of an asynchronous BCI. IEEE Trans. Neural Syst. Rehabil. Eng.12, 258-65.

Vidaurre, C., Krämer, N., Blankertz, B., Schlögl, A., 2009. Time domain parameters as a feature for EEG-based brain computer interfaces. Neural Networks, 22, 1313-1319.

Walter, W.G., Cooper, R., Aldridge, V.J., McCallum, W.C., Winter, A.L., 1964. Contingent negative variation: An electric sign of sensorimotor association and expectancy in the human brain. Nature 25:380–384.

Wang, Y., Wang, R., Gao, X., Hong, B., Gao S., 2006. A practical VEP-based brain-computer interface. IEEE Transactions on Neural Systems and Rehabilitation Engineering 14(2):234–239.

Wang, Y., Zhang, Z., Li, Y., Gao, X, Gao, S., Yang, F., 2004. BCI competition 2003–data set iv: an algorithm based on CSSD and FDA for classifying single-trial EEG. IEEE Trans. Biomed. Eng., 51, 1081–1086.

Weiskopf, N., Scharnowski, F., Veit, R., Goebel, R., Birbaumer, N., Mathiak, K., 2004. Self-regulation of local brain activity using real-time functional magnetic resonance imaging (fMRI). Journal of Physiology, Paris 98(4–6):357–373.

Weitkunat, R., 1991. Digital Biosignal Processing. New York, Elsevier.

Wessberg J., Stambaugh C.R., Kralik J.D., Beck P.D., Laubach M., Chapin J.K., Kim J., Biggs S.J., Srinivasan M.A., Nicolelis M.A., 2000. Real-time prediction of hand trajectory by ensembles of cortical neurons in primates. Nature 408(6810), 361-365.

Williams, J.C., Rennaker, R.L., Kipke, D.R., 1999. Long-term neural recording characteristics of wire microelectrode arrays implanted in cerebral cortex. Brain Res. Brain Res. Protoc., 4, 303-313.

Wilson, J.A., Felton, E.A., Garell, P.C., Schalk, G., Williams, J.C., 2006. ECoG factors underlying multimodal control of a brain-computer interface. IEEE Transactions on Neural Systems and Rehabilitation Engineering 14(2):246–250.

Wolpaw, J.R., McFarland, D.J., 2004. Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. Proceedings of the National Academy of Sciences (PNAS) 101(51):17849–17854.

Wolpaw, J.R., McFarland, D.J., Vaughan, T.M., Schalk, G., 2003. The Wadsworth center brain-computer interface (BCI) research and development program. IEEE Trans. Neur. Sys. Rehab. Eng., 11, 1–4.

Wolpaw, J.R., Birbaumerc, N., McFarlanda, D.J., Pfurtschellere, G., Vaughana, T.M., 2002. Brain–computer interfaces for communication and control. Clin. Neurophysiol., 113, 767-791.

Wolpaw, J.R., McFarland D.J., Neatb, G.W., Forneris, C.A., 1991. An EEG-based brain-computer interface for cursor control. Electroencephalogr Clin Neurophysiol 78(3): 252-259.

Wolpert, D.H., 1992. Stacked generalization. Neural Netw., 5, 241–259.

Wu, W., Gao, Y., Bienenstock, E., Donoghues, J.P., Black, M.J., 2006. Bayesian Population Decoding of Motor Cortical Activity Using a Kalman Filter. Neural Computation, 18(1), 80-118.

Yates, F., 1933. The analysis of replicated experiments when the field results are incomplete. The empire journal of experimental agriculture, 1, 129.

Yom-Tov, E., Inbar, G.F., 2003. Detection of movement-related potentials from the electro-encephalogram for possible use in a brain-computer interface. Medical and Biological Engineering and Computing 41(1), 85-93.

Zaveri, H.P., Williams, W.J., Iasemidis, L.D., Sackellares, J.C., 1992. Time-frequency representation of electrocorticograms in temporal lobe epilepsy. IEEE Transactions on Biomedical Engineering, 39(5), 502-509.

Zhang, H., Guan, C., 2010. A maximum mutual information approach for constructing a 1D continuous control signal at a self-paced brain-computer interface. J. Neural Eng., 7(2010), pp. 1-11.

Zhao, Q., Caiafa, C.F., Cichocki, A., Zhang, L., Phan, A.H., 2009. Slice Oriented Tensor Decomposition of EEG Data for Feature Extraction in Space, Frequency and Time Domains. Lecture Notes in Computer Science, 5863, 221-228.

Zhao, Q., Zhang, L., Cichocki, A., 2008. EEG-based asynchronous BCI control of a car in 3D virtual reality environments. Chinese Science Bulletin, 54(1), 78-87.

Zhu J., Yao, T., 2004. An evaluation of statistical spam filtering techniques. ACM Transactions on Asian Language Information Processing (TALIP), vol. 3, no.4, pp. 243-269.

Zou, H., Hastie, T., 2005. Regularization and variable selection via the elastic net. J. Roy. Statist. Soc. Ser. B 67, 301-320.